Accelerating CVA and CVA Sensitivities using Quasi-Monte Carlo Methods

Stefano Renzitti^{*}, Pouya Bastani[†], and Steven Sivorot[‡]

IHS Markit Canada, 1066 West Hastings St Suite 1780, Vancouver, BC V6E 3X1, Canada

October 6, 2020

This is the peer reviewed version of the following article: Renzitti, S., Bastani, P., and Sivorot, S.. 2020. Accelerating CVA and CVA Sensitivities Using Quasi-Monte Carlo Methods. Wilmott Magazine 108, 7893, which has been accepted in final form at https://doi.org/10.1002/wilm.10860.

Abstract

We compare the efficiency of quasi-Monte Carlo (QMC) methods to classical Monte Carlo (MC) method and MC with antithetic sampling in computing credit valuation adjustment (CVA) and CVA sensitivities for various portfolios of interest rate swaps using a multicurrency extension to the Hull-White model. For uncollateralized portfolios using local models, we find that QMC with Sobol sequences and the Brownian bridge discretization can produce results as accurate as classical MC with 10,000 simulations when using on average roughly only 800 simulations, a speed-up by a factor of 12. However, we also find that the acceleration varies significantly across portfolios (increasing with moneyness and usually, but not always, decreasing with the number of factors), calculation types (order from highest to lowest, usually, but not always, CVA and CR Deltas, IR and FX Deltas, and IR and FX Vegas), and the choice of model (local models usually outperform global models). While the Brownian bridge discretization is less effective on the collateralized portfolios, the so-called Brownian bridge portfolio interpolation technique significantly improves the results. Randomization of Sobol' sequences, a technique shown to increase the convergence rate of QMC on a particular class of integrands, is found to be most effective on test cases with small numbers of dimensions.

Keywords: CVA, Greeks, Monte Carlo, Quasi-Monte Carlo, Sobol' Sequences

1 Introduction

One of the most important counterparty credit risk measures is the credit valuation adjustment (CVA), defined as the present value of the potential loss due to a counterparty failing to meet their contractual obligations. Risk neutral pricing states that the present value is equal to the expected value of the payoff using risk adjusted probabilities. The CVA payoff is the netted portfolio value less collateral (floored at zero) at the time of counterparty default, multiplied by one minus the recovery rate. The payoff is at counterparty level, potentially path dependent (collateral, early exercise conditions, lags between fixings and cash flows), and subject to change.

^{*}stefano.renzitti@ihsmarkit.com

 $^{^{\}dagger}$ pouya.bastani@ihsmarkit.com

 $^{^{\}ddagger}$ steven.sivorot@ihsmarkit.com

The expectation of high-dimensional, fluid payoffs of this sort are in practice estimated with Monte Carlo (MC) simulation (see Gregory (2015) [11]).

Monte Carlo estimation of an expectation involves randomly sampling the payoff n times according to the risk neutral probabilities and averaging the results. The estimate approaches the true expectation with probability 1 with a normally distributed error with zero mean and standard deviation equal to the standard deviation of the payoff (a constant) divided by the square root of the number of replications n used (see Jäckel (2002) [13] and Glasserman (2004) [10] for detailed explanations of MC and its application to finance).

Requiring the error to be on average 100 times smaller than the standard deviation of the CVA payoff requires 10,000 replications, a number typically used. This highlights the main disadvantage of MC: its computational expense. This is of particular importance in the context of CVA where each evaluation of the payoff is also computationally expensive. Consider a bank with 100,000 trades that uses 200 exposure dates in the time discretization. One replication of the CVA payoffs across all counterparties requires roughly 10,000,000 trade prices (assuming trade maturities are evenly distributed) and thus one MC CVA estimate using 10,000 paths requires of the order of 100,000,000,000 trade price evaluations. Furthermore, many banks risk manage these credit adjustments, and to do so requires the calculation of the derivatives of the CVA with respect to the market prices of the instruments used to hedge it. Bump and run techniques require at least one full MC CVA calculation per derivative. 200 derivatives bring the computational load up to 20,000,000,000 trade price evaluations per day.

Not surprisingly, quants have been searching for ways to accelerate this massive calculation. One successful line of research uses adjoint algorithmic differentiation (AAD) to compute the derivatives, reducing the computational burden to a fixed multiple (5 to 10 times depending on the problem and memory handling) of the baseline CVA calculation, no matter how many derivatives are required (see Giles and Glasserman (2006) [9] and Capriotti et al. (2011) [7] for more information). Assuming a conservative fixed multiple of 10, this would reduce the total number of calculations by a factor of 20, requiring 1,000,000,000 trade price evaluations. This dramatic improvement, however, does not come for free. The implementation of an AAD enabled system requires large changes to existing code libraries, requiring a significant upfront investment to implement. As a consequence, many still compute the derivatives using bump and run techniques.

In another line of research, Ghamami and Zhang (2014) [8] highlight that direct and independent simulation of the portfolio value to each time step, rather than the usual chronological time stepping scheme, diversifies the errors across each CVA time bucket, leading to a significant reduction of the standard error of the final sum across time. The benefit of the direct simulation approach is reduced if simulating to each time step independently is more computationally expensive than simulating to each step sequentially using a common simulation path. Highly path dependent simulation models and portfolios may not benefit as a result, but the technique looks quite promising for portfolios of uncollateralized vanillas.

In a similar line of research, Burnett et al. (2016) [5] note that the computational expense of calculating valuation adjustment risks (derivatives) vary significantly across different counterparties, and that the computational expense is uncorrelated with the size of the adjustment error. This opens up the possibility to optimally allocate computational resources where they are needed most, using a different number of paths and/or time steps for different counterparties and risks. They formalize this idea by setting up and minimizing the expected unexplained Profit and Loss (PnL) by varying the number of paths and frequency of time steps allocated to each counterparty and risk, subject to a computational time constraint. The acceleration they report computing FVA on a sample Barclays portfolio is impressive, roughly in line with the acceleration provided by AAD.

In this article, we explore yet another, potentially complementary, acceleration technique commonly used to price single trade payoffs called quasi-Monte Carlo (QMC). The mechanics are identical to classical Monte Carlo simulation with the exception that the pseudo random numbers (PRN) are replaced with carefully selected low-discrepancy sequences (LDS) that are more evenly distributed. This has been shown to result in convergence rates between $\mathcal{O}(n^{-\frac{1}{2}})$ and $\mathcal{O}(n^{-1})$, depending on the complexity of the payoff, and more specifically, on how heavily the payoff depends on interacting terms between the uniform random variables. Payoffs that heavily depend on interaction terms result in convergence rates closer to classical Monte Carlo, $\mathcal{O}(n^{-\frac{1}{2}})$, whereas payoffs that do not result in convergence rates closer to $\mathcal{O}(n^{-1})$. The degree of interaction is called the *effective dimension*, and was first described by Caflisch et al. (1997) [6]. In the same paper, they also point out that the effective dimension of a payoff is not fixed, and can be reduced by reformulating the payoff or risk factor simulation to depend more on fewer, earlier indexed random variables. A common example is the *Brownian bridge discretization*.

The best-case convergence rate would be a remarkable result, reducing the number of required replications by a factor of $1/\sqrt{n}$, an acceleration of 100 times in our example above. The obvious question is then can we achieve the optimal convergence rate when estimating CVA payoffs, despite their high *nominal dimension*? This is hard to answer in general, as it depends on the portfolio (which varies by counterparty and by bank) and the model used to evolve the risk factors. In this paper, we narrow the scope of this question and focus on the efficiency of QMC when used to compute CVA and CVA sensitivities for a specific class of portfolios (vanilla interest rate swap portfolios) and a specific model class (multi-currency extension to the Hull-White model, see Hull and White (1994) [12], with deterministic hazard rates).

We do not present any new theories or methodologies in this paper, the core concepts that we use have been described in detail in papers and books referenced throughout. Our primary contribution is the presentation of how well QMC methods work when used to estimate CVA and CVA sensitivities for a specific class of portfolios and models. To the best of our knowledge, aside from the work of Bianchetti et al. (2016) [3] and the brief analysis at the end of Sobol' et al. (2012) [29], very few results have been published on this topic.

Overviews of the payoffs, simulation models, and numerical techniques are provided first. The core set of numerical experiments and the corresponding results are presented second. We then look to see how these results are impacted by various changes, including using global models rather than local models, using more simulation time steps, and finally adding collateral.

2 Monte Carlo CVA Estimate

The bilateral CVA to a portfolio held against a counterparty c is

$$CVA_{t_0} = (1 - R_c) \int_{t_0}^T \mathbb{E}_{t_0} \left[\frac{N_{t_0}}{N_{t+\delta}} \max(V_{t+\delta} + F_{t,t+\delta} - C_{t+\delta}, 0) \right] P(\tau^b > t) dP(\tau^c < t), \quad (1)$$

where t_0 is today, T is the maturity date of the portfolio, R_c is the recovery rate if the counterparty defaults (assumed to be deterministic), N_t is the numeraire corresponding the risk adjusted probability measure used in the expectation, δ is the time lag between the default event and the settlement of the portfolio (margin period of risk), $V_{t+\delta}$ is the netted idealized value of all trades held against the counterparty c at time $t + \delta$, $F_{t,t+\delta}$ is the netted primary trade flows not paid between the default time t and settlement time $t + \delta$ with accrued interest, $C_{t+\delta}$ is the collateral posted by the counterparty (positive) or bank (negative) at settlement time $t + \delta$, τ_c and τ_b are the default times of the counterparty and bank, and $P(\tau_c < t)$ is the time t_0 probability τ_c is less than t, and $P(\tau_b > t)$ is the time t_0 probability τ_b is greater than t (we have have assumed a reduced form model where default times are independent of each other and the exposures). See Gregory (2015) [11] for a general description and Anderson et al. (2017) [1] for the nuances of δ , F and C. The uncollateralized case with zero settlement lag

simplifies to:

$$CVA_{t_0} = (1 - R_c) \int_{t_0}^T \mathbb{E}_{t_0} \left[\frac{N_{t_0}}{N_t} \max(V_t, 0) \right] P(\tau^b > t) dP(\tau^c < t).$$
(2)

Given a time grid $t = [t_0, t_1, t_2, \ldots, t_m]'$, a left-hand Riemann-sum approximation¹ to the integral in (2) and a Monte Carlo approximation of the expectation using *n* simulations results in the following formula for the CVA estimate:

$$CVA_{t_0}^{MC} = \frac{(1 - R_c)}{n} \sum_{i=0}^{m-1} \sum_{\omega=0}^{n-1} \frac{N_{t_0}}{N_{t_i+\delta}^{\omega}} \max(V_{t_i+\delta}^{\omega} + F_{t_i,t_i+\delta}^{\omega} - C_{t_i+\delta}^{\omega}, 0) P(\tau^b > t_{i+1})P(t_i \le \tau^c < t_{i+1}), \quad (3)$$

where the superscript ω denotes the path index for stochastic variables. These variables are deterministic functions of the realization of an $n^{\texttt{factor}}$ -dimensional risk factor vector X_t at a set of fixing dates on or before their respective observation dates.

$$N_{t_i+\delta}^{\omega} = \mathbf{N}(t_i + \delta, X_{s \le t_i+\delta}^{\omega})$$

$$V_{t_i+\delta}^{\omega} = \mathbf{V}(t_i + \delta, X_{s \le t_i+\delta}^{\omega})$$

$$C_{t_i+\delta}^{\omega} = \mathbf{C}(t_i + \delta, X_{s \le t_i+\delta}^{\omega}, V_{s \le t_i+\delta}^{\omega})$$

$$F_{t_i,t_i+\delta}^{\omega} = \mathbf{F}(t_i, t_i + \delta, X_{s \le t_i+\delta}^{\omega})$$
(4)

We consider a model with the following risk factor dynamics:

$$dX_t = [AX_t + f_t] dt + \Sigma_t dW_t \tag{5}$$

where A is a n^{factor} by n^{factor} constant matrix of reals, f_t is a time dependent vector of n^{factor} reals, Σ_t is a time dependent matrix of n^{factor} by n^{factor} reals, and dW_t is a vector of n^{factor} Wiener processes in the domestic T-forward measure, with a constant instantaneous correlation matrix R.

The risk factor vector X_t is simulated to a set of fixing dates $t^f = [t_0^f, t_1^f, t_2^f, \ldots, t_{n^{\text{fixing}}}^f]'$, where $t_0^f = 0 < t_1^f < t_2^f < \ldots < t_{n^{\text{fixing}}}^f = T$ (which includes the exposure dates and settlement dates $t, t + \delta \in t^f$) using, for example, a basic Euler scheme². In equations, start at $t_0^f = 0$ with X_0 given, and chronologically iterate over the fixing indices $i = 1, 2, \ldots, n^{\text{fixing}}$:

$$\begin{aligned} \xi_{i}^{\omega} &= \Phi^{-1}(\mathfrak{u}_{i}^{\omega}) \\ \hat{W}_{t_{i}^{f}}^{\omega} &= \hat{W}_{t_{i-1}^{f}}^{\omega} + \xi_{i}^{\omega} \sqrt{\Delta_{i}} \\ W_{t_{i}^{f}}^{\omega} &= W_{t_{i-1}^{f}}^{\omega} + \sqrt{R} \left[\hat{W}_{t_{i}^{f}}^{\omega} - \hat{W}_{t_{i-1}^{f}}^{\omega} \right] \\ X_{t_{i}^{f}}^{\omega} &= X_{t_{i-1}^{f}}^{\omega} + \left[A X_{t_{i-1}^{f}}^{\omega} + f_{t_{i-1}^{f}} \right] \Delta_{i} + \Sigma_{t_{i-1}^{f}} \left[W_{t_{i}^{f}}^{\omega} - W_{t_{i-1}^{f}}^{\omega} \right] \end{aligned}$$
(6)

where u_i^{ω} is the realization of an n^{factor} -dimensional vector of uniformly distributed random variables in [0, 1) in path ω , $\Phi^{-1}(x)$ is the inverse cumulative standard normal distribution function operating element wise on x, ξ_i^{ω} is the realization of an n^{factor} -dimensional vector of standard normal random variables in path ω , \hat{W}_t^{ω} is the realization of an n^{factor} -dimensional

 $^{^{1}}$ Note that other higher order numerical integration schemes such as trapazoidal or Simpson's rule can instead be used, but for simplicity and ease of notation they are not explored here.

 $^{^{2}}$ An exact stepping scheme exists for this system of SDEs, as described in Ng, Peterson, and Rodriguez (2011) [23]. However, we use the Euler scheme for the simplicity of exposition and to localize dependencies of the risk factors to bumps to model parameter volatilities.

vector of independent Wiener processes at time t and in path ω with $\hat{W}_0 = 0, 0$ is an n^{factor} dimensional vector of zeros, $\Delta_i = t_i^f - t_{i-1}^f$, W_t^{ω} is the realization of an n^{factor} -dimensional vector of correlated Wiener processes as defined above in path ω , and \sqrt{R} is the n^{factor} by n^{factor} matrix containing the square root of the correlation matrix in the eigen decomposition sense³. In a pathwise simulation, the portfolio values and collateral balances are computed in step with the risk factors.

Define **X** as the n^{factor} by n^{fixing} matrix of risk factor values at all future required time steps $\mathbf{X} = [X_{t_0^f}, X_{t_1^f}, \dots, X_{t_{n_{\text{fixing}}}}]$ and $\boldsymbol{\xi}$ as the stacked $d = n^{\text{factor}} n^{\text{fixing}}$ -dimensional vector of standard normal random variables $\boldsymbol{\xi} = [\boldsymbol{\xi}_1^T, \boldsymbol{\xi}_2^T, \dots, \boldsymbol{\xi}_{n_{\text{fixing}}}^T]^T$. Define the function from the normal random variables $\boldsymbol{\xi}$ to the risk factor values **X** as $\mathbf{X} = g(\boldsymbol{\xi})$, where we suppress the dependence on the model parameters A, f and Σ , and initial risk factor values X_0 for notational convenience. Write the CVA payoff in equation (3) as a function π of the risk factor matrix $\mathbf{X}, \pi(\mathbf{X}) \equiv \sum_{i=0}^{m-1} \pi_i(\mathbf{X})$ and $\pi_i(\mathbf{X}) \equiv (1 - R_c) \frac{N_{t_0}}{N_{t_i+\delta}} \max(V_{t_i+\delta} + F_{t_i,t_i+\delta} - C_{t_i+\delta}, 0)P(\tau^b > t_{i+1})P(t_i \leq \tau^c < t_{i+1})$. Now, by expressing the standard normal variables in terms of *d*dimensional vector of uniform random variables $\mathbf{u} = [u_1^T, u_2^T, \dots, u_{n_{\text{fixing}}}]^T$ we can explicitly approximate CVA as a (Riemann type) sum over the unit hypercube $[0, 1)^d$:

$$CVA_{t_0} \approx CVA_{t_0}^{MC} = \frac{1}{n} \sum_{\omega=0}^{n-1} \pi(g(\Phi^{-1}(\mathbf{u}^{\omega}))).$$
(7)

Thus the computation of CVA reduces to an integration problem over the unit hypercube $[0, 1)^d$. In this paper we discuss and compare random, deterministic, and hybrid sampling methods to generate realizations of the risk factor matrix **X**. It turns out that, for some payoffs, through a suitable choice of the map $\mathbf{u} \to \mathbf{X}$, the (effective) dimensionality of the integration problem can be reduced to less than the nominal dimension d. Such a choice can benefit QMC in particular, leading to faster and/or more accurate CVA estimates. The remainder of this section introduces the various MC concepts under investigation in more detail.

2.1 Classical Monte Carlo

In classical MC \mathbf{u}^{ω} is generated to be independent from each other and all earlier draws using a pseudo random number generator, such as the Mersenne Twister algorithm (Matsumoto and Nishimura (1997) [21]). The strong law of large numbers guarantees that if the payoff is integrable the approximation approaches the true answer as n approaches ∞ with probability 1. If the payoff is additionally square integrable, the standard deviation of the MC estimate is equal to the standard deviation of the payoff divided by the square root of the number of replicas n:

$$\sigma_{\text{MC,n}} = \sigma_{\text{Payoff}} n^{-\frac{1}{2}}.$$
(8)

Furthermore, the central limit theorem guarantees that the distribution of the MC estimate converges to the normal distribution as the number of replicas approaches ∞ , \sqrt{n} (CVA^{MC} – CVA) $\xrightarrow{d} \mathcal{N}(0, \sigma_{\mathsf{Payoff}}^2)$, where $\mathcal{N}(\mu, \sigma^2)$ is the normal distribution with mean μ and variance σ^2 . This allows us to place confidence intervals around the MC estimates. In practice, $\sigma_{\mathsf{Payoff}}^2$ is not known, but can be easily estimated along with the mean using the sample variance: $\sigma_{\mathsf{Payoff}}^2 \approx \frac{1}{n-1} \sum_{\omega=0}^{n-1} \left(\pi(g(\Phi^{-1}(\mathbf{u}^{\omega}))) - \text{CVA}^{\text{MC}} \right)^2$.

³Principal Component Analysis (PCA) can be applied as an additional step to reduce the dimensionality of the problem, which is expected to yield improvements for the QMC methods. In our test cases, none of the eigenvalues were small enough to merit such a reduction.

2.2 Classical Monte Carlo with Antithetic Sampling

Classical Monte Carlo with antithetic sampling (AMC) attempts to reduce the variance of the estimator by introducing negative dependence between pairs of random replications. Assuming n to be an even number, draw n/2 uniformly distributed random variables \mathbf{u}^{ω} for $\omega = 0, 1, \ldots, n/2 - 1$ and take the mirror image of each to obtain the antithetic samples $1 - \mathbf{u}^{\omega}$, also valid uniformly distributed random variables. Appending the two sequences together results in n samples, where each of the antithetic pairs is perfectly negatively correlated.

In order to use the standard error formula (8) that applies only to n independent samples, we can average the payoff of the antithetic pairs first to obtain n/2 independent samples, and then average these to obtain the AMC estimate:

$$CVA_{AMC} = \frac{1}{n} \sum_{i=0}^{n/2-1} \pi(g(\Phi^{-1}(\mathbf{u}_{AMC}^{i}))) + \pi(g(\Phi^{-1}(1-\mathbf{u}_{AMC}^{i})))$$
(9)

The standard deviation of the AMC estimate is then given by a formula analogous to (8) but with a reduced set of replications:

$$\sigma_{\text{AMC,n}} = \sqrt{2}\sigma_{\text{Pairs}} n^{-\frac{1}{2}}.$$
(10)

Comparing equations (8) and (10), it is clear that the convergence rate is unchanged. However, the constant $\sqrt{2}\sigma_{\text{Pairs}}$ is smaller than σ_{payoff} if the payoffs evaluated over the antithetic pairs are negatively correlated.

2.3 Quasi-Monte Carlo

Instead of attempting to generate a sequence of pseudo-random numbers in $[0, 1)^d$ that tends to cluster, QMC methods use *low-discrepancy sequences* (LDS) $\{\mathbf{u}_{\text{LDS}}^{\omega}\}_{\omega=0}^{n}$ that are constructed to fill the unit hypercube more evenly. The remaining mechanics of the QMC method are identical to those of classical Monte Carlo:

$$CVA^{QMC} = \frac{1}{n} \sum_{\omega=0}^{n-1} \pi(g(\Phi^{-1}(\mathbf{u}_{LDS}^{\omega})))$$
(11)

A result based on the Koksma-Hlawka inequality states that the integration error of QMC methods is proportional to $\ln^d(n)/n$. Thus, for low-dimensional problems, QMC can provide a quadratic speed-up compared to MC methods, but the minimum number of points required to achieve this accelerated convergence rate grows exponentially with d, a bound that can be prohibitively high for high-dimensional integrands. It turns out, however, that even for many high-dimensional problems QMC methods can still achieve a higher convergence rate than classical MC with a feasible number of sample paths. In fact, what has been repeatedly observed is that the expected error (see Section 2.5) follows a power law [2, 6, 10],

$$\sigma_{\text{QMC,n}} \approx \alpha n^{-\beta},\tag{12}$$

where β , estimated empirically, is found to be close to 1 for many integrands. This accelerated convergence rate is often attributed to the low so-called *effective dimension* of the problem⁴, which is found to be a better predictor of the QMC error bound that the nominal dimension d

⁴Without getting too technical, the effective dimension d^e describes the (effective) dependence of a function $f(x_1, ..., x_d)$ on its nominally d dimensional argument. It can be viewed as the maximum dimensionality of function arguments appearing in a sufficiently accurate low order analysis of variance (ANOVA) expansion of f when viewed as a random function. Sometimes a coordinate transformation $g(y_1, ..., y_d) = f(h(y_1, ..., y_d))$ can help reduce the effective dimension of the problem at hand. Suffice to say that $0 \le d^e \le d$. See Caflisch et al. (1997) [6] and appendix D for more details.

(see Kucherenko and Shah (2007) [16] and Kucherenko et al. (2011) [17]). It turns out that for many payoffs encountered in practice, especially in financial engineering, the effective dimension is or can be reduced to be significantly less than the nominal dimension.

Of the low-discrepancy sequences available, Sobol sequences (Sobol' (1967) [28]) have been shown to have superior convergence properties, relatively robust uniformity properties across dimensions, and are very efficient to generate (see Glasserman (2004) [10], and Jackel (2002) [13] for example). Sobol sequences, however, are not unique and require a set of direction numbers to initialize the sequence. The choice of direction numbers greatly affects the efficiency of the method (Jackel (2002) [13]). We use direction numbers provided by BRODA [4] commercially that allow for the generation of sequences in up to 65,536 dimensions, sufficient for most practical applications in financial engineering. The direction numbers were chosen such that the resulting sequence satisfies Sobol's uniformity property A and any projection onto five adjacent dimensions also satisfies Sobol's uniformity porperty A'. Property A states that if we split each [0, 1)-interval of the d-dimensional unit hypercube into two equal intervals, i.e. [0, 0.5) and [0.5, 1), making 2^d bins, then the first 2^d elements of the sequence will fall into unique bins, the next 2^d elements fall into unique bins, etc. Property A' is similar, but rather than subdividing each interval equally into two, we subdivide them equally into four.

2.4 Quasi-Monte Carlo and Brownian Bridge

The effective dimension of a problem is not fixed and can be reduced by reformulating the payoff and/or risk factor simulation. The Brownian bridge path construction is a common example, one that has been shown to reduce the sensitivity of the effective dimension to the number of time steps used in the simulation and is thus a promising technique for CVA. The Brownian bridge discretization makes use of the Brownian bridge formula:

$$\hat{W}_{t} = \hat{W}_{s} + \frac{t-s}{u-s} \left(\hat{W}_{u} - \hat{W}_{s} \right) + \xi_{t} \sqrt{\frac{(t-s)(u-t)}{u-s}}$$
(13)

where s < t < u and $\xi_t \sim \mathcal{N}(0, 1)$, to simulate each independent Brownian motion in a nonchronological order such that each step explains a maximum amount of remaining variation. Assume we need the n^{factor} -dimensional vector of Brownian motions at M equal time steps, and that M is a power of 2. The initial value of $W_0 = \mathbf{0}$. Next, we generate \hat{W}_M as $\hat{W}_M = \sqrt{T}\xi_0$ where ξ_0 is an n^{factor} -dimensional vector of normal independent random variables. Formula (13) is then used to create $W_{M/2}$ from W_0 , W_M and ξ_1 , then $W_{M/4}$ from $W_{M/0}$, $W_{M/2}$, and ξ_2 , then $W_{3M/4}$ from $W_{M/2}$, W_M , and ξ_3 , and so on. See Caflisch et al. (1997) [6] for the original description. Each successive random draw adds finer and finer details to the path, increasing the dependence on fewer and earlier indexed uniform variables, and reducing the effective dimension for payoffs that primarily depend on the overall shape of the Brownian path.

Once obtained, the n^{factor} by n^{fixing} matrix of Brownian motions \hat{W} are used in a regular chronological simulation of the risk factors. Iterate from $i = 1, 2, \ldots, n^{\text{fixing}}$:

$$W_{t_i^f} = W_{t_{i-1}^f} + \sqrt{R} \left[\hat{W}_{t_i^f} - \hat{W}_{t_{i-1}^f} \right]$$
(14)

$$X_{t_i^f} = X_{t_{i-1}^f} + \left[AX_{t_{i-1}^f} + f_{t_{i-1}^f}\right] \Delta_i + \sum_{t_{i-1}^f} \left[W_{t_i^f} - W_{t_{i-1}^f}\right]$$
(15)

We define the end-to-end risk factor simulation using the Brownian bridge discretization function from the normal random variables $\boldsymbol{\xi}$ to the future risk factor values \mathbf{X} as $\mathbf{X}_{BB} = BB(\boldsymbol{\xi})$. The Brownian bridge QMC CVA estimate is then equal to:

$$CVA^{QMC,BB} = \frac{1}{n} \sum_{\omega=0}^{n-1} \pi(BB(\Phi^{-1}(\mathbf{u}_{LDS}^{\omega})))$$
(16)

Other Brownian path constructions are available, including PCA of the entire risk factor system for all fixing dates [10, 13], but we do not explore them further in this paper.

2.5 QMC Error Estimates

In the classical MC framework, the standard error over n paths is given by equation (8) or (10), depending on whether the method of antithetic variates is used or not. In the QMC framework, on the other hand, as a result of the inter-dependence between consecutive elements of the low-discrepancy sequence, no formula of practical use exists for calculating the standard error from the sample payoff replications.

Instead, it is usually estimated empirically (see Caflisch et al. (1997) [6] and Bianchetti et al. (2015) [2]) by evaluating the CVA k times using n non-overlapping elements of the low-discrepancy sequence, and taking the standard deviation of these k CVA estimates $\text{CVA}_{0,n}^{\text{QMC}}$, $\text{CVA}_{1,n}^{\text{QMC}}$, ..., $\text{CVA}_{k-1,n}^{\text{QMC}}$ where $\text{CVA}_{j,n}^{\text{QMC}} = \frac{1}{n} \sum_{\omega=j}^{j+n-1} \pi(g(\Phi^{-1}(\mathbf{u}_{\text{LDS}}^{\omega})))$. The empirical expected error, or root mean squared error (RMSE), using n simulations is then defined as:

$$\hat{\sigma}_{\text{QMC},n} = \sqrt{\frac{1}{k-1} \sum_{j=0}^{k-1} \left(\text{CVA}_{j,n}^{\text{QMC}} - \text{CVA} \right)^2}$$
(17)

where CVA is the "exact" value, estimated with a very large number of simulation paths n_{∞} . The procedure is repeated l times using $n_0 < n_1 < \ldots < n_{l-1}$ simulations (all powers of two to ensure smooth convergence of Sobol' sequences) to obtain estimates for $\hat{\sigma}_{\text{QMC},n_0}$, $\hat{\sigma}_{\text{QMC},n_1}$, \ldots , $\hat{\sigma}_{\text{QMC},n_{l-1}}$. Then, assuming a power law of convergence (see equation (12)), the parameters α and β are estimated using linear least squares regression of the natural log of the empirical error onto the natural log of the paths⁵. The same procedure can be used to compute the expected error of classical MC with and without antithetic sampling, but now each sample would be independent and normally distributed for larger n. This estimate approaches the analytic results in equations (8) or (10) as the number of trials reaches infinity.

2.6 Randomized Quasi-Monte Carlo

Randomized quasi Monte Carlo (RQMC) is a technique for obtaining independent estimates from multiple QMC runs that can be used to estimate the standard deviation of the error and also potentially increase the optimal convergence rate (for an overview see Glasserman (2004) [10]). A low-discrepancy sequence $\{\mathbf{u}_{\text{LDS}}^{\omega}\}_{\omega=0}^{n-1}$ is randomized k times to produce k independent sequences $\{\mathbf{y}_{j}^{\omega}\}_{i=0}^{n-1}$ of uniformly distributed points in $[0,1)^{d}$, where $j = 0, 1, \ldots, k-1$. Performing QMC evaluations over k independent trials using these sequences leads to estimates $\text{CVA}_{0,n}^{\text{RQMC}}$, $\text{CVA}_{1,n}^{\text{RQMC}}$, ..., $\text{CVA}_{k-1,n}^{\text{RQMC}}$ where $\text{CVA}_{j,n}^{\text{RQMC}} = \frac{1}{n} \sum_{\omega=0}^{n-1} \pi(g(\Phi^{-1}(\mathbf{y}_{j}^{\omega})))$. The standard deviation of the error when using n simulations is then equal to:

$$\hat{\sigma}_{\text{RQMC},n} = \sqrt{\frac{1}{k-1} \sum_{j=0}^{k-1} \left(\text{CVA}_{j,n}^{\text{RQMC}} - \text{CVA} \right)^2}$$
(18)

where CVA can be set to the "exact" value, estimated with a very large number of simulation paths n_{∞} as before, or to the mean of the k estimates. Proceeding as in the previous section, standard deviation estimates $\hat{\sigma}_{RQMC,n_0}, \hat{\sigma}_{RQMC,n_1}, \ldots, \hat{\sigma}_{RQMC,n_{l-1}}$ for difference numbers of simulation paths $n_0 < n_1 < \ldots < n_{l-1}$ can be utilized to identify the power law dependence of the RQMC

⁵According to this definition, we know that $\hat{\sigma}_{QMC,1}$ will converge to the payoff standard deviation as k approaches infinity. An alternative modelling choice is then to impose this restriction on the convergence formula, and only solve for β using linear regression with a known intercept of the log of the payoff standard deviation.

error on the simulation path number. Least squares regression is used to estimate the power law parameters describing the error as a function of the number of paths.

Several techniques exist for randomizing $\{\mathbf{u}_{\text{LDS}}^{\omega}\}_{\omega=0}^{n-1}$ to obtain $\{\mathbf{y}_{j}^{\omega}\}_{\omega=0}^{n-1}$ for $j = 0, 1, \ldots, k-1$. We focus our attention on the *linear permutation of digits* method by Matousek (1998) [20]. As a more implementation-efficient approximation to the Owen (1997) [25] full scrambling algorithm, the linear permutation of digits maps the binary representation of $x = 0.a_1a_2...$ to $y = 0.b_1b_2...$ using

$$b_l = \sum_{k=1}^{l} h_{kl} a_k + g_l \mod 2$$
 (19)

with h_{kl} and g_l chosen randomly and independently from $\{0, 1\}$ and $h_{ll} = 1$. Note that, for a particular dimension, the same permutation parameters are used across all the elements in the sequence.

A remarkable result by Owen (1997) [25] is that for sufficiently smooth integrands, RQMC using a particular class of randomization schemes, among which is the linear permutation of digits, can lead to integration errors of order $\mathcal{O}(n^{-3/2})$. Although the smoothness condition of Owen's theorem does not hold in most derivative pricing applications (Glasserman (2004) [10]), and the theoretical enhanced convergence rate may not kick in for feasible sizes of n, it has been observed to improve the convergence rate in several financial applications (Caflisch et al. (1997) [6], and Tan and Boyle (2000) [30] for example).

3 Numerical Experiments

We create uncollateralized portfolios with 1, 6, and 11 single currency pay fixed receive float swaps, each swap in a particular portfolio being in a unique currency. The foreign notionals are set to the corresponding spot FX rate as of the simulation start date. All swap legs have a one year frequency. We create several variations of each portfolio, varying the moneyness from far-in-the-money (fixed rates set to par minus 300bp), in-the-money (fixed rates set to par minus 100bp), at-the-money, out-of-the-money (fixed rate set to par plus 100bp), to far-out-ofthe-money (fixed rate set to par plus 300bp). All trades mature in ten years.

Closed form solutions are used to value the swaps forward in time as a function of the zero bonds and exchange rates, which are simulated based on a cross currency extension to the Hull-White model [12]. The short rate in each currency is controlled by one mean reverting normally distributed factor and the exchange rate between a foreign currency and the domestic currency is governed by a log-normal random variable. Credit events are assumed to be independent of the exposure profiles and numeraire, and thus do not directly affect the simulation. See section 2 for details. In total, the 1-currency portfolio depends on 1 factor, the 6-currency portfolio on 11 factors, and the 11-currency portfolio on 21 factors. Exposure dates are quarterly spaced, with additional exposures added the day of and the day after the underlying swaps' cash flows.

The model is calibrated to market data as of August 1st, 2016. The mean reversions for each short rate process are explicitly input at 0.03. The short rate volatility in each currency is parameterized as piecewise constant and calibrated to a diagonal of co-terminal swaptions from an at-the-money (ATM) swaption volatility matrix that matches the maturity of the swap being analyzed. The discontinuities of the short rate volatility are equal to the swaption maturities. The FX volatility is also parameterized as piecewise constant and calibrated to a strip of ATM FX options. The discontinuities of the FX volatility are set equal to the expiry of the FX options. Correlations are estimated historically.

We compute CVA level and sensitivities to parallel shifts of the curves. For each currency (except the domestic), we compute four sensitivities: IR Delta (a parallel one basis point shift to the yield curve), IR Vega (a parallel one percent shift to the swaption volatility matrix), FX Delta (a one tenth of a percent shift to the spot FX rate), and FX Vega (a parallel one percent

shift to the implied FX volatility curve). Last, we compute CR Delta (a parallel shift of one basis point to the credit curve), assumed to be the same for all portfolios. Portfolios depending on a different number of currencies consequently require a different number of sensitivities: the 1-currency domestic portfolios require 3 sensitivities (1 CR Delta, 1 IR Delta, and 1 IR Vega), the 6-currency portfolios require 23 sensitivities (1 CR Delta, 6 IR Deltas, 6 IR Vegas, 5 FX Deltas, and 5 FX Vegas), and the 11-currency portfolios require 43 sensitivities (1 CR Delta, 11 IR Deltas, 11 IR Vegas, 10 FX Deltas, and 10 FX Vegas). Results are presented as the difference from the base, which when divided by the corresponding bump size are equal to the forward finite difference estimates of the first derivatives.

As outlined above, we apply several Monte Carlo simulation schemes, using a basic Euler risk factor stepper, to estimate CVA and CVA sensitivities. To summarize, the five Monte Carlo variations are as follows:

- 1. Classical Monte Carlo (MC): Employs the Mersenne Twister pseudo random number generator to produce n uniformly distributed random vectors. See section 2.1 for details.
- 2. Classical Monte Carlo with Antithetic Sampling (AMC): Employs the Mersenne Twister pseudo random number generator to produce n/2 uniformly distributed random vectors (*n* even). Another set of n/2 uniformly distributed random vectors are formed via reflection of the first set. See section 2.2 for details.
- 3. **Quasi-Monte Carlo (QMC):** Utilizes Sobol' sequences with BRODA direction numbers [4] to generate *n* low-discrepancy vectors. See section 2.3 for details.
- 4. Quasi-Monte Carlo + Brownian Bridge (QMC+BB): Utilizes Sobol' sequences with BRODA direction numbers to generate n vectors of low-discrepancy vectors, and the Brownian bridge discretization to map these to vectors of independent Wiener processes. See section 2.4 for details.
- 5. Randomized Quasi-Monte Carlo + Brownian Bridge (RQMC+BB): Similar to QMC+BB but now using randomized Sobol' sequences based on the linear permutation of digits from Matoušek (1998) [20]. See section 2.6 for details.

To provide a succinct measure of comparison we estimate for each of the portfolios, calculation types, and the five Monte Carlo variants, the expected number of paths needed to match the expected error of classical MC when using 10,000 paths. We refer to this measure as the *equivalent paths*⁶. It is found by assuming the error for each payoff and methodology follows a power law of convergence, the parameters of which are estimated empirically. Using these equations we solve for the number of paths that are required to match the classical MC error with 10,000 paths (see section A for details). From the equivalent paths measure we also calculate the acceleration factor as

acceleration factor =
$$\frac{10,000}{\# \text{ of equivalent paths}}$$
. (20)

4 Results

In this section we compare the performance of various Monte Carlo methods (AMC, QMC, QMC+BB, and RQMC+BB) for calculating CVA and CVA sensitivities on various portfolios of 10-year payer interest rate swaps⁷ in 1, 6, and 11 currencies; there is one swap in each currency,

 $^{^6{\}rm For}$ simulation methods involving QMC, i.e. QMC, QMC+BB, and RQMC+BB, it is recommended that the number of paths to use be a power of 2.

 $^{^{7}}$ Note that the CVA results on the portfolio of payer swaps can equivalently be viewed as DVA results on the portfolio of receiver swaps.



Figure 1: Acceleration factor as a function of the swap spread (in bp) above par rate. The calculations involve CVA level and sensitivities for a 1-currency portfolio containing one 10-year payer swap in each currency. Local model results are shown for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).

with the swap rates set to par plus a given spread that is varied from -300bp (creating a farin-the-money portfolio) to 300bp (creating a far-out-of-the-money portfolio). In general, for the *n*-currency portfolio there is 1 CVA level and CR Delta calculation, *n* IR Delta and Vega calculations, as well as n - 1 FX Delta and Vega calculations, for a total of 4n calculations. When showing the sensitivity results for the multi-currency portfolios, the acceleration factor is obtained from the averaged equivalent paths over the 2n - 1 IR/FX Delta and the 2n - 1IR/FX Vega results, with equal weighting given to each.

Figure 1 shows that, for CVA level and CR Delta calculations on the 1-currency far-inthe-money portfolio, AMC and QMC both have roughly equal acceleration factor between 30 and 40, but as the portfolio grows more out-of-the-money, QMC performance deteriorates less quickly than AMC, so that with the swap rate set to par plus 300bp, QMC needs roughly half the number of paths as AMC and classical MC. Using Brownian bridge to reduce the effective dimension of the problem, QMC+BB and RQMC+BB achieve a significantly higher acceleration rates of 50 and 150, respectively, as well as slower deterioration rates. The clear improvement in efficiency across the QMC methods for in-the-money portfolios may be caused by the payoff becoming more separable as flooring the portfolio at zero rarely happens, reducing the effective dimension. This lowered effective dimension also seems to allow randomization to further increase the efficiency of the simulation. We note that the increased complexity of IR Delta and Vega calculations results in slower convergence rate across all methods.

Figure 2 shows the acceleration factor for CVA level and sensitivity results across the different simulation methods for the 6-currency portfolio. Here, QMC offers little or no advantage



Figure 2: Acceleration factor as a function of the swap spread (in bp) above par rate. The calculations involve CVA level and sensitivities for a 6-currency portfolio containing one 10-year payer swap in each currency. Local model results are shown for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).

over AMC: as the portfolio moneyness decreases, the acceleration of the two schemes decreases roughly equally. As in the single-currency case, RQMC+BB offers a higher acceleration rate than QMC+BB for the far-in-the-money portfolio (acceleration factor of 80 vs 40 for the CVA level and CR Delta), but as the portfolio moneyness decreases the two methods perform at par (with a more modest acceleration factor of 4 or equivalent paths of 2500). On average, across the 22 IR/FX Delta calculations, QMC+BB and QMC+BB have an acceleration factor of 30 and 40, respectively, when the swaps' fixed rates are set to par minus 300bp, but decreases to 4 when the spread is increased to 300bp. For the IR/FX Vega calculations, the acceleration factor of the two methods starts at 20 and 30, respectively, for the far-in-the-money portfolio (spread of -300bp) and decreases to 3 for both as the swaps' spread is increased to 300bp.

As we increase the portfolio size to include 11 swaps in 11 currencies, naturally the dimension of the problem increases. As illustrated in Figure 3 QMC under-performs AMC across all payoffs; in fact, the acceleration factor is under 1 for the out-of-the-money portfolios, meaning that QMC requires more than 10,000 paths to match classical MC's error at 10,000 paths. One such example is the CR Delta calculation at 0 strike, requiring over 14,000 paths to reach MC's error at 10,000 paths (acceleration factor of 0.7). As shown in Figure 4 for CR Delta calculations, the convergence plot of QMC for this payoff exhibits a plateau whereby as the number of paths in the QMC calculation is increased from 512 to 8192, the standard error of the simulated payoff decreases at a rate much lower than the rate implied by the first half of the plot as the paths are repeatedly doubled from 2 to 512. However, as the number of paths is increased from 8192 to 16384, the error dramatically drops. This effect, we hypothesize,



Figure 3: Acceleration factor as a function of the swap spread (in bp) above par rate. The calculations involve CVA level and sensitivities for an 11-currency portfolio containing one 10-year payer swap in each currency. Local model results are shown for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).

is caused by the "holes" in certain lower-dimensional projections of the Sobol' sequences (see Appendix B) that have a period of 8192. In other words, this plateauing effect hints at the existence of regions of the unit hypercube that do not get uniformly filled, even at 8192 points; for this particular payoff these regions seem to be important.

As for the smaller portfolio sizes, Brownian bridge results in a significant boost: for level and CR Delta calculations the acceleration factor starts out at roughly 30 and 50 for QMC+BB and RQMC+BB, respectively, in the far-in-the-money case at spread of -300bp, decreasing to 4 for both methods as the spread increases to the far-out-of-the-money limit of 300bp. For IR/FX Delta and Vega calculations, the randomization makes little difference, most likely due to increased dimension of the problem. The Brownian bridge discretization, however, leads to a speed-up of 20 (12) times for the IR/FX Delta (Vega) calculations in the most optimal case, but only an acceleration factor of 3 (2) in the worst case.

In summary, the CVA level and sensitivity results on the portfolio of payer swaps considered in this section show that as the portfolio moneyness increases, the acceleration factor generally increases across all simulation methods⁸. We believe this is caused by the payoff becoming more symmetric and linear as the moneyness increases, since maxing at 0 is triggered less often in equation (1). In these results we observe that QMC with Brownian bridge discretization provides a significant advantage over AMC and the standard QMC. Randomization can provide an additional boost, but most notably for the smaller in-the-money portfolios, while for larger,

 $^{^{8}}$ For DVA results on the same portfolio of payer swaps, the scenario is mostly reversed: the acceleration factor generally decreases across all methods as the moneyness increases.



Figure 4: Standard error vs number of paths for the CR Delta calculation over the 11-currency portfolio.

more out-of-the-money portfolios the performance of RQMC+BB is at par with or slightly worse than the non-randomized version, QMC+BB. Generally, the acceleration increases as the portfolios grow more in the money, with IR/FX Delta and Vega calculations requiring on average more paths than level and CR Delta calculations to reach the same level of accuracy.

Figure 5 compares the acceleration factor across the different random number generators as the number of currencies c is increased, which in effect increases the problem dimension by a factor of 2c - 1. As these plots show, for the payoffs considered here, the performance of standard QMC can be very sensitive to the number of dimensions in the problem: for the calculations on the single-currency portfolio QMC can lead to a significant speed-up over AMC; however, with increasing number of currencies, this performance gain deteriorates quickly, so that in the 11-currency portfolios QMC repeatedly under-performs AMC.

One way to lower the error in the QMC estimate (without using additional samples) is to reformulate the problem so that the dependence of the integrand (payoff) on the random variables is changed in such a way that the effective dimension of the problem is reduced. Brownian bridge discretization achieves this by generating the Brownian path in a different order⁹ than the standard chronological one. As illustrated in Figure 5, with increasing portfolio size, the acceleration factor of QMC+BB and RQMC+BB on the CVA level and CR Delta calculations deteriorates by a much smaller factor compared to standard QMC; randomization offers some advantage, mainly on smaller portfolios.

The results discussed above apply to portfolios simulated using the local model. In practice, a global model may be used instead, in which the number of simulated risk factors and the simulation horizon is typically independent of particular trades, but depend on the make-up of the entire portfolio. The next section discusses the impact of the various Monte Carlo approaches in the global model setting.

4.1 Global Models

For operational reasons and potentially entity level calculations (required to compute some flavours of FVA), it is common to use a *global model*, capable of simultaneously dealing with all counterparty netting sets. Denote the number of factors needed for the global model as n_g and the number of factors needed for the counterparty as n_c , where we usually have $n_c \ll n_g$. In

⁹Other orderings are possible, see for instance Lin and Wang (2008) [19].



Figure 5: Acceleration factor as a function of the number of currencies in the portfolio, containing one ATM 10-year payer swap in each currency. Local model results are shown for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).

such cases, only n_c risk factors from X_t are used in a particular counterparty simulation, but if a global model is used, each of these depends on the weighted sum of n_g random variables per time step. In a 30-currency ($n_g = 59$) global model as used for our test cases, the 1-, 6-, and 11-currency portfolios all depend on the same sequence of high-dimensional random vectors, but with different weights, which results in increased the effective dimension of the problem, and degraded performance across all QMC methods.

Another aspect of the global model is that it uses a common time horizon T across all netting sets, which in practice is chosen to be the largest of the maturity dates. To analyze the effect of switching from a local to global model, we set T = 50 years in our test cases, keeping all trades the same as before: one 10-year payer swap in each currency. This increased time horizon has two implications.

Firstly, since the risk factors are simulated in the T-forward measure, increasing T from 10 to 50 years changes the distribution of the numeraire and in turn that of the normalized payoff¹⁰, which affects the performance of AMC as well as all QMC methods. We note that

¹⁰The variance in the distribution of the numeraire, i.e. discount factor D(t, T), is zero at $t = t_0$ and t = T, and achieves its maximum at some point $t \in (t_0, T)$. Changing the time horizon T alters the distribution of the numeraire, and depending on the nature of the payoff (e.g. CVA vs DVA or a payer swap vs receiver swap) and its correlation with the numeraire, this change can increase (decrease) the variation in the normalized payoff, resulting in increase (decrease) in the number of simulation paths required to achieve a given level of accuracy. In fact, while for AMC and QMC the acceleration factor for CVA results generally deteriorate with increasing time horizon, we found DVA calculations on the same portfolio of payer swaps exhibiting accelerated convergence. Results for QMC methods involving Brownian bridge across CVA (DVA) calculation also generally deteriorate (improve) with increasing time horizon, though the pattern is more erratic. We believe this is due to the fact that varying T results in different indices of Sobol' dimensions, having different projection properties, being used

another choice of numeraire that removes this dependence, such as the risk-neutral measure, may be a better candidate for a global model simulation¹¹.

Secondly, since the Brownian bridge discretization depends on the simulation horizon T, switching to the global model would mean that, on average, higher number of random variables, and later-indexed dimensions of the Sobol's sequence are used to simulate the Wiener process at each point in the (0, 10yr) time frame. Both these effects are expected to decrease the acceleration factor of these methods compared to the local model.



Figure 6: Acceleration factor as a function of the swap spread (in bp) above par rate. The calculations involve CVA level and sensitivities for a 1-currency portfolio containing one 10-year payer swap in each currency. Local and global model results are shown in dotted and solid lines, respectively, for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).

Figures 6–8 show the CVA level and sensitivity results on the 1-, 6-, and 11-currency portfolios, respectively, calculated using this global model (solid lines). Comparing these to the local model results (dotted lines), we see that the acceleration factor of all methods decreases as we switch from the local to the global model, with the change being more pronounced on the smaller-size portfolios. The degradation in AMC acceleration is solely attributed to the use of the *T*-forward measure and the payoff becoming less symmetric as *T* is increased from 10 years (in the local model) to 50 years (in the global model). QMC performance, however, has deteriorated the most; it often under-performs AMC and even classical MC in some cases (acceleration factor lower than 1). This is true even in the single-currency portfolio in which

to calculate the Wiener process for the same time frame, e.g. (0, 10yr).

¹¹Our simulation results with the risk-neutral measure show that for CVA calculation, under the risk-neutral measure, in contrast to the T-forward measure, AMC and standard QMC are statistically unaffected by purely changing the change in the time horizon. Under such a measure, the deterioration in switching from a local to global model would solely be attributed to the increase in the number of factors as the number of currencies simulated is increased to 30.



Figure 7: Acceleration factor as a function of the swap spread (in bp) above par rate. The calculations involve CVA level and sensitivities for a 6-currency portfolio containing one 10-year payer swap in each currency. Local and global model results are shown in dotted and solid lines, respectively, for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).

QMC outperformed AMC by a significant margin in the local model. In fact, as depicted in Figure 6, in the far-out-of-the-money portfolio, with the swaps' fixed rate at par plus 300bp, QMC's acceleration is below 0.7, requiring roughly 15,000 equivalent paths for CVA level and sensitivities.

As alluded to above, the reason behind this degradation in QMC performance is two-fold: increasing the time horizon T from 10 to 50 years combined with the use of T-forward measure, as well as simulating 30 currencies (instead of 1, for instance, in the single currency portfolio) directly contributes to a significant increase in the number of problem dimensions. As a result, the adjacent dimensions of the Sobol' sequence that have a desirable uniformity property¹² are not utilized effectively to simulate interdependent factors in the model.

In addition to the above two factors, QMC+BB and RQMC+BB performance is further influenced negatively by the increase in simulation time horizon as more and later Sobol' dimensions are used to simulate the portion of the Brownian motion within the 10-year period (e.g. the first two random variables simulate W_{50yr} and W_{25yr} in the global model instead of W_{10yr} and W_{5yr} in the local model).

As illustrated in Figure 9, AMC results are insensitive to the number of currencies in the portfolio. This is because the changes in weights assigned to the random variables is inconsequential for MC, as the convergence rate is fixed at 1/2. The acceleration factor for QMC, QMC+BB, and RQMC+BB also remains roughly unchanged for different-size portfolios as the

¹²The BRODA[4] SobolSeq65536 was constructed such that each adjacent set of five dimensions satisfies property A', see Sobol' et al. (2012) [29].



Figure 8: Acceleration factor as a function of the swap spread (in bp) above par rate. The calculations involve CVA level and sensitivities for an 11-currency portfolio containing one 10-year payer swap in each currency. Local and global model results are shown in dotted and solid lines, respectively, for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).

changes in weights is not expected to significantly alter the effective dimension of the problem. We believe any observed difference is due to chance in the sense that the different set of weights used for the 1-, 6-, and 11-currency portfolios causes a particular dependence on the set of dimensions of the Sobol' sequence that can have different uniformity properties. We note that in the global model the 1-trial randomization in RQMC+BB does not appear to offer any advantage over QMC+BB.

4.2 Number of Time Steps

We now explore the impact of using more time steps in the risk factor simulation while holding the number of exposure dates fixed. Using a local model, we calculate CVA of a portfolio consisting of a single ATM 10-year payer swap, and increase the number of time steps from 66 to 3,653. The net effect is to use more random variables, increasing the nominal dimension of the problem.

As Figure 10 shows, while AMC is unsurprisingly unaffected by the increase in the number of time steps, QMC's performance steadily diminishes. Note, however, that as the number of time steps increases from 750 to 3650 the acceleration factor for QMC decreases by a relatively small amount. This indicates that the number of time steps does not correlate directly with the number of effective dimensions of the problem, even for QMC. Impressively, QMC+BB shows little dependence on the number of time steps, while the results for RQMC+BB are rather erratic.



Figure 9: Acceleration factor as a function of the number of currencies in the portfolio, containing one ATM 10-year payer swap in each currency. Local and global model results are shown in dotted and solid lines, respectively, for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).



Figure 10: Acceleration factor as a function of the number of time steps across various simulation methods: AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple). The calculation involves CVA calculation for a portfolio containing one ATM 10-year payer swap using a local model.

5 Collateralized Counterparties

Consider the CVA to a collateralized counterparty with daily margining, zero thresholds and zero minimum transfer amounts. Assume no flows are paid between the counterparty default date τ_c and the portfolio settlement date $\tau_c + \delta$ and that collateral and cash are invested in the numeraire asset:

$$CVA = (1 - R_c) \int_0^T \mathbb{E}_0 \left[\frac{N_0}{N_{t+\delta}} \max\left(V_{t+\delta} + \sum_{t \le t_j^c < t+\delta} \frac{N_{t+\delta}}{N_{t_j^c}} CF_{t_j^c} - \frac{N_{t+\delta}}{N_t} V_t, 0 \right) \right] P(\tau^b > t) dP(\tau^c < t), \quad (21)$$

where CF_t are the netted primary portfolio cash flows at time t, and $\{t_j^c\}$ are the cash flow dates. We consider two methods for estimating this CVA: 1) margin period of risk adjusted exposure dates, discussed in Section 5.1, and 2) portfolio interpolation using Brownian bridge, discussed in Section 5.2.

5.1 Margin Period of Risk Adjusted Exposure Dates

In this method we evaluate the portfolio at the original exposure dates t_i as before, but now in addition we evaluate the portfolio value at the portfolio settlement dates $t_i^s = t_i + \delta$:

$$CVA \approx \frac{N_0}{n} (1 - R_c) \sum_{i=0}^{m-1} \sum_{\omega=0}^{n-1} \left[\max\left(\frac{V_{t_i+\delta}^{\omega}}{N_{t_i+\delta}^{\omega}} + \sum_{t_i \le t_j^c < t_i+\delta} \frac{CF_{t_j}^{\omega}}{N_{t_j}^{\omega}} - \frac{V_{t_i}^{\omega}}{N_{t_i}^{\omega}}, 0 \right) \right]$$

$$P(\tau^b > t_{i+1})P(t_i \le \tau^c < t_{i+1}). \quad (22)$$

This is similar to the *classical* margin period of risk model with a coarse grid look back discussed in Andersen et al. (2017) [1], but here we use a coarse grid look forward. Results for portfolios of 1, 6, and 11 currencies containing one ATM 10-year payer swap in each currency and a margin period of risk δ of 10 business days are presented in Figure 11.



Figure 11: Acceleration factor as a function of the number of currencies in the portfolio, containing one ATM 10-year payer swap in each currency. The portfolio is fully collateralized, has a 10 business day margin period of risk, and is valued on both the exposure dates and the margin period adjusted dates. Local and global model results are shown in dotted and solid lines, respectively, for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).

For the local model results (shown in dotted lines), standard QMC (with chronological discretization) outperforms all other methods in the single-currency case; results for larger portfolios are more or less the same across the different QMC schemes. Comparing this figure with Figure 5, we note that, in the local model, standard QMC performs much better on collateralized portfolios than without.

This can occur because rather than the exposure primarily depending on the portfolio value (as in the uncollateralized case), the collateralized exposure primarily depends on the difference in the portfolio value over the margin period of risk. For the single-currency single-factor case with the chronological discretization, the difference is captured by one Brownian increment, which is computed with a simple inverse cumulative normal transform of one Sobol' variable. Since the marginal of all Sobol' variables are perfectly uniform when using binary segments of the sequence (see Appendix B), this portion of the payoff is computed optimally. On the other hand, for the uncollateralized case, the portfolio value depends on the weighted sum of the inverse cumulative normal transform of several Sobol' variables from time 0 to t, the distribution of which may be highly non-uniform. For the 6- and 11-currency portfolios, in the collateralized case, it is the weighted sum of the adjacent 11 and 21 dimensions, respectively, of the Sobol' sequence that is used to calculate the difference in the portfolio value over the margin period of risk. Thus, the number of dimensions involved in the sum is still lower than that of the uncollateralized portfolio. This could explain why standard QMC performs better on the collateralized portfolios in the local model.

We also note that Brownian bridge discretization can actually increase the effective dimension of payoffs, such as collateralized derivatives that depend on the Brownian increments. But as is clear from equation (21), the calculation of the CVA depends also on the numeraire level value. Hence, while Brownian bridge negatively influences the calculation of the terms within the max function, it improves the calculation of the numeraire. In addition, as explained above, the effectiveness of QMC on simulating the collateralized payoffs decreases with increasing number of currencies as the payoff becomes dependent on more adjacent Sobol' dimensions. As a result, we see from Figure 11 that while QMC outperforms QMC+BB on the 1-currency portfolios in the local model, the situation reverses for the higher dimension portfolios. We note that this is inline with the equivalence principle of Wang and Sloan (2001) [31], i.e the performance of the Brownian bridge or any other discretization scheme depends on the particular payoff it is applied to.

While Brownian bridge discretization negatively affects the collateralized portfolio results in the local model, the method still proves effective in the global model case: both QMC+BB and RQMC+BB perform roughly at par across the different portfolio sizes and outperform all other simulation methods. Standard QMC results are at par or worse than AMC (and even classical MC in some cases); we believe this is because the special A' uniformity property that holds across every adjacent 5-dimension of the (BRODA) Sobol' sequence (see Sobol' et al. (2012) [29]) cannot be utilized as effectively as in the local model to estimate the payoff.

5.2 Portfolio Interpolation using Brownian Bridge

Equation (22) requires potentially twice as many portfolio evaluations as an uncollateralized portfolio and thus can take twice as long to compute. An optimization, described by Pykhtin (2009) [27] and similarly by Andersen et al. (2017) [1], avoids the additional valuations by using a Brownian bridge to interpolate the portfolio value at time $t + \delta$ from the simulated portfolio values at the original time grid t. In addition to the computational savings by reducing the number of portfolio evaluations, this method has the added benefit that a large portion of the payoff variation can be described by one random variable per time step, an important step to improve the effectiveness of QMC methods. To this end, rewrite the CVA payoff of equation (21) in terms of a martingale:

$$CVA = (1 - R_c) \int_0^T \mathbb{E}_0 \left[\frac{N_0}{N_t} \max\left(M_{t+\delta}^t - M_t^t, 0 \right) \right] P(\tau^b > t) dP(\tau^c < t),$$
(23)

where

$$M_{t+\delta}^t = \frac{N_t}{N_{t+\delta}} V_{t+\delta} + \sum_{t \le t_j^c < t+\delta} \frac{N_t}{N_{t_j^c}} \mathrm{CF}_{t_j^c}.$$

Simulate the martingale to the original exposure dates t, and then use a Brownian bridge interpolation technique to find the martingale realizations at the settlement dates $t + \delta$. If we assume the volatility of the martingale is piece-wise constant and deterministic over each time interval¹³, it can be estimated from the sample standard deviation of the martingale increment, denoted as $\hat{\sigma}_i^M$, and CVA can be estimated as:

$$CVA \approx \frac{N_0}{n} (1-R) \sum_{i=0}^{m-1} \sum_{\omega=0}^{n-1} \left[\frac{1}{N_{t_i}^{\omega}} \max\left(\Delta M_{t_i+\delta}^{t_i,\omega}, 0\right) \right] P(\tau^b > t_{i+1}) P(t_i \le \tau^c < t_{i+1})$$
(24)

 $^{^{13}}$ The variance of the martingale over time period *i* could be estimated by regressing the square of the change over that time bucket onto a suitable basis expansion of the value of the martingale itself at the left hand side of the interval, although, in the test cases we consider in this paper, we find the assumption of state-independent volatility to be reasonable.

where

$$\Delta M_{t_{i+\delta}}^{t_{i,\omega}} = \frac{\left(M_{t_{i+1}}^{t_{i,\omega}} - M_{t_{i}}^{t_{i,\omega}}\right)\delta}{\Delta_{i}} + \hat{\sigma}_{i}^{M}\Phi^{-1}(\mathbf{u}_{i}^{\omega})\sqrt{\frac{\delta(\Delta_{i} - \delta)}{\Delta_{i}}},\tag{25}$$

 u_i^{ω} is a scalar uniform random variable for path ω and time bucket *i*, independent of all other uniform random variables used to simulate the risk factors underlying the model, and $\Delta_i = t_{i+1} - t_i$. In all Monte Carlo methods we take the *m* additional uniform dimensions from the end of the sequence, after all uniforms required for the risk factor simulation have been used.



Figure 12: Acceleration factor as a function of the number of currencies in the portfolio, containing one ATM 10-year payer swap in each currency. The portfolio is fully collateralized, has a 10 business day margin period of risk, and is valued only on the exposure dates and uses the *Brownian Bridge* technique to estimate value on the margin period adjusted dates. Local and global model results are shown in dotted and solid lines, respectively, for AMC (blue), QMC (red), QMC+BB (yellow), and RQMC+BB (purple).

Test results for the same set of portfolios as before but using the Brownian bridge portfolio interpolation method are presented in Figure 12 for the local (dotted) and global models (solid). Comparison with Figure 11 shows that this technique significantly improves the QMC acceleration factor so that it now outperforms AMC on most test cases. QMC+BB and RQMC+BB results are improved to a lesser degree, while AMC, as expected, is unaffected by this technique.

6 Conclusion

Our results above show that QMC works very well on small portfolios with in-the-money trades and over relatively simple payoffs that are inherently of lower effective dimension compared to more complex calculations (i.e. sensitivity) over larger-size portfolios containing out-of-themoney trades. This finding is consistent with conclusions of Morokoff and Caflisch (1995) [22] on the performance degradation of QMC with growing dimensions in the problem.

Overall, QMC with the Brownian bridge discretization (QMC+BB) provides a significant boost over both classical MC and MC with antithetic sampling (AMC), in particular on portfolios of uncollateralized swaps, offering roughly an acceleration factor of 12 over the classical MC counterpart in the cases tested (results based on average over all payoffs, spreads, and number of currencies in the portfolio using the local model). The size of the improvement varies significantly between portfolios, calculation types, and simulation models. With the global model, in general, the efficiency of the methods involving QMC decreases as the adjacent dimensions of the Sobol' sequence cannot be used as effectively. In particular, the acceleration of QMC+BB decreases from 12 to 3 when switching to a 30-currency global model with a 50-year simulation horizon. Obtaining the optimal acceleration required reformulating the payoff to depend on fewer and earlier indexed elements of the quasi-uniform variables (see Glasserman (2004) [10]).

The Brownian bridge risk factor discretization is less effective when used with collateralized portfolios, where the margin period of risk dominates the exposure. In such cases the payoff depends more on the difference between the portfolio values at adjacent exposure dates rather than the level, reducing the acceleration factor from 12 to 7 when using the local model. When using the global model, on the other hand, the Brownian bridge discretization, on average, performs equally on collateralized and uncollateralized portfolios, with an acceleration factor of 3. The Brownian bridge portfolio interpolation technique is shown to further increases the efficiency of QMC+BB to 5 across the global model results. Moreover, randomization (RQMC+BB) can provide additional speed-up, especially on lower-dimensional problems.

We note that while the randomization scheme used in this work is based on the *linear* permutation of digits introduced by Matousek (1998) [20], other mechanisms are possible. In particular, our future work aims at investigating the effect of scrambled nets (see Owen (1998) [26]) in further accelerating RQMC performance for CVA and CVA sensitivity calculations. The potentially large number of permutations required for this form of scrambling may require an optimized implementation of the original algorithm.

While we have shown that the Brownian bridge discretization can vastly improve the convergence rate of the standard QMC method, we are interested in investigating other methods of Brownian path construction, such as those based on the principal component analysis (see Glasserman (2004) [10]) or the alternative Brownian bridge construction of Lin and Wang (2007) [19], under both the T-forward and risk-neutral measure.

Other potential ideas for future research include combining QMC methods with optimizations discussed in the introduction. Combining, for example, the direct and independent simulation approach of Ghamami and Zhang (2014) [8] with randomized QMC methods with the Brownian bridge can diversify the errors across the time bucket, just as with pseudo random numbers, but in addition can reduce the number of uniforms needed to describe the payoff. Another promising direction is to combine the work of Burnett et al. (2016) [5] with the estimated QMC convergence constants and rates for the various netting sets and risks to find the optimal number of QMC paths needed for each risk. We expect the additional variability in convergence rates may make this type of analysis even more important, as paths and time spent can yield vastly different benefits.

We have shown that QMC with the Brownian bridge discretization is suitable to the estimation of CVA and CVA sensitivities when using a cross currency, single factor per currency Hull-White model when the portfolios contain vanilla interest rate swaps. Large variability in the acceleration was found, with the general pattern that more complex payoffs had lower acceleration factors. The way the acceleration decreased with the complexity was in some cases smooth, especially when going from very small number of dimensions to a moderate number of dimension, but was quite noisy when going from a moderate number of dimension to a high number of dimensions. This variability and noise make it difficult to extrapolate the results to different models and portfolios. It is extremely important then for potential users to try QMC methods out on their portfolio and set of risk calculations. It has been our experience that this work is worthwhile, yielding benefits that are maximized when local models are deployed.

Appendices

A Equivalent Paths

To investigate the convergence pattern of the different schemes, we use the methodology outlined in section 2.5. We start by calculating a total of 1,638,400 payoff values. These values are split up into multiple trials that each contain n subsetted paths. The subset of path combinations that we chose to summarize over vary from 2 to 16,384 in powers of 2, i.e. $n_i = 2^i$ for $i = 1, 2, \ldots, 14$. We average over n_i paths for each of the $k_{n_i} = (100 \times 16,384)/n_i$ consecutive trials, so that at 16,384 paths 100 trials are generated, while at 2 paths 819,200 trials are used. These averages can then be used to determine the error estimate, with the methodology varying depending on the random number type.

For the randomized scheme, RQMC+BB, rather than generating 1,638,400 paths, we only generate the first n_i paths. The sequence is independently randomized k_{n_i} times in order to obtain 1,638,400 payoff values. The averages are then calculated with the same methodology as in the other schemes.

Once the root mean squared error (RMSE) has been calculated for each number of paths, we regress the log RMSE onto the log number of paths to estimate the convergence coefficients α and β . In the end, for each method \mathcal{M} and payoff f, the expected error is approximated by:

$$\sigma(\mathcal{M}, f, n) = \alpha(\mathcal{M}, f) n^{-\beta(\mathcal{M}, f)}$$
(26)

Finding the number of paths p using method \mathcal{M}_1 required to match the expected error using n paths using method \mathcal{M}_2 is then found by equating the two convergence equations and solving for p:

$$p(\mathcal{M}_1, f, \mathcal{M}_2, n) = \left(\frac{\alpha(\mathcal{M}_2, f)}{\alpha(\mathcal{M}_1, f)}\right)^{-\frac{1}{\beta(\mathcal{M}_1, f)}} n^{\frac{\beta(\mathcal{M}_2, f)}{\beta(\mathcal{M}_1, f)}}$$
(27)

In this paper we primarily set $\mathcal{M}_2 = MC$ and n = 10,000. The acceleration factor a is simply n/p:

$$a(\mathcal{M}_1, f, \mathcal{M}_2, n) = \left(\frac{\alpha(\mathcal{M}_2, f)}{\alpha(\mathcal{M}_1, f)}\right)^{\frac{1}{\beta(\mathcal{M}_1, f)}} n^{1 - \frac{\beta(\mathcal{M}_2, f)}{\beta(\mathcal{M}_1, f)}}$$
(28)

B Low-dimensional projections of the Sobol' sequence

The Sobol' sequence is an example of the so-called low-discrepancy sequences, a term traditionally reserved for *d*-dimensional sequences $x_0, x_1, \ldots, x_{n-1}$ that have a star discrepancy bound of (see Glasserman (2004) [10])

$$D_n^*(x_1,\ldots,x_n) = \mathcal{O}\left(\frac{\ln^d(n)}{n}\right).$$
(29)

In contrast, random numbers on average have a discrepancy bound of (Niederreiter (1992) [24])

$$D_n^*(x_1, \dots, x_n) = \mathcal{O}\left(\frac{\sqrt{\ln(\ln(n))}}{\sqrt{n}}\right).$$
(30)

The Koksma-Hlawka inequality, which gives an upper bound on the Monte Carlo approximation error as a multiple of the underlying sequence's star discrepancy, combined with equation (30) gives the well-known $\mathcal{O}(1/\sqrt{n})$ convergence rate of classical MC. On the other hand, since the logarithmic term $\ln^d(n)$ can be absorbed into any power of n, equation (29) implies that for low enough dimensions QMC can achieve a convergence rate of $\mathcal{O}(1/n^{1-\epsilon})$ for all $\epsilon > 0$. However, as the dimensionality increases, the number of points required to obtain this superior convergence rate increases dramatically.

Yet in practice, it is sometimes found that QMC outperforms standard MC even for highdimensional problems and relatively low path numbers. As suggested in [6, 16, 17], the efficiency of QMC for a particular problem is better attributed to its effective dimension than nominal dimension.

For instance, the function

$$f(\mathbf{x}) = \sum_{i=1}^{d} \sin(4\pi x_i)^2, \ \mathbf{x} = [x_1, \dots, x_d]^T \in [0, 1)^d.$$

has nominal dimension d, but an effective dimension of only 1. In other words, the integral of f can be calculated as the sum of integrals of functions of only 1 dimensions.

Since the binary segments of the Sobol' sequence have perfectly uniform 1-dimensional projections¹⁴ (see Figure 13a), they can be used to provide highly accurate approximations of integrals of functions whose effective dimension is or can be reduced to 1 (e.g. with a Brownian bridge discretization the effective dimension of a European call option price can be reduced to 1 (see Bianchetti et al. (2015) [2]). Thus, QMC combined with Brownian bridge (i.e. QMC+BB) yields a highly accurate approximation for this payoff. This is in contrast to classical MC based on pseudo-random numbers for which no particular segment of the sequence can be guaranteed to have a perfectly uniform distribution (Figure 13b).



Figure 13: 32-bin histogram of the first 32 points of a Sobol' sequence (a) and Mersenne Twister sequence (b) in the [0, 1) interval.

While each dimension of the Sobol' sequence has perfectly uniform distribution, the same cannot be said of its higher-dimensional projections. For instance, while a set of 512 points of a particular Sobol' sequence in dimensions i and j each have perfectly uniform distribution, the 2-dimensional projection of these points on the (i, j) plane has regions devoid of any points (Figure 15a). In this scenario, an integrand of effective dimension 1, e.g. $f(\mathbf{x}) = \sin(4\pi x_{47}) + \sin(4\pi x_{49})$ (Figure 14a), would have much smaller integration error compared to another one with effective dimension 2, such as $g(\mathbf{x}) = \sin(4\pi x_{47}) \cdot \sin(4\pi x_{49})$ (Figure 14b), even though both have a

¹⁴Each binary segment $\{\mathbf{x}_i : j2^m < i < (j+1)2^m\}, j = 0, 1, 2, \dots$ of the Sobol' sequence is a (t, m, d)-net for any m > t (see Glasserman (2004) [10]). With t = 0 and d = 1, this implies that exactly $2^t = 1$ point falls in each elementary subinterval of volume $2^{t-m} = 2^{-m}$, resulting in perfectly uniform 1-dimensional histograms.

nominal (i.e. total) dimension of 2, where $\mathbf{x} = [x_{47}, x_{49}]^T$. This is because $f(\mathbf{x})$ depends on marginal distributions that are of one dimension, whereas $g(\mathbf{x})$ depends on the two-dimensional distribution of the sequence on the (47,49)-plane. In fact, because $g(\mathbf{x})$ is non-negative only in the non-void squares of size $1/4 \times 1/4 = 1/16$ in Figure 15a, calculation of the integral of this function based on the first 512 points of the Sobol' sequence would be integrating only the positive regions of the integrand, resulting in a highly inaccurate approximation of the true value of the integral, which we know to be 0. On the other hand, with 1024 points of the same sequence a much more uniform distribution of the points on the (47,49)-plane is obtained (Figure 15b), resulting in a much more accurate approximation of the integral of $g(\mathbf{x})$.



Figure 14: Plot of 2-dimensional functions $f(\mathbf{x})$ and $g(\mathbf{x})$ on the unit (47,49)-plane with effective dimensions of 1 and 2, respectively.



Figure 15: 2-dimensional projection of the BRODA Sobol' sequence (SobolSeq65536) onto the (47, 49)-plane.

Thus, as the number of points is increased (while working solely with binary segments of size 2^m), generally the distribution of points on the lower-dimensional projections improves and the number of dimension pairs that exhibit such "hole" structures decreases. Nevertheless, we have found that undesirable 2D projections can still persist even at high number of paths. For instance, projection of the first $2^{13} = 8192$ points of the BRODA[4] Sobol' sequence (SobolSeq65536) onto the (62,73)-plane fills only two of the quadrants on the plane (Figure 16a). Similarly, the projection of the first $2^{10} = 1024$ points of the Joe/Kuo Sobol' sequence

(D6) (see Joe and Kuo (2008) [14, 15]) on to the (401,358)-plane fills the same two quadrants only (Figure 16b).



Figure 16: The projection of the first 8192 points of the BRODA Sobol' sequence (SobolSeq65536) onto the (62,73)-plane (left) and the first 1024 points of the Joe/Kuo Sobol' Sequence (D6) onto the (358, 401)-plane (right).

An equivalent way to characterizing these patterns is by noting that, in both cases, all the points are either less than 0.5 or bigger to equal to 0.5 across the aforementioned dimension pairs, i.e. they all share the same most-significant-bit (MSB). Such coincidence stems from the fact that the direction numbers used to generate both of the dimensions have the same MSB. Specifically, across dimensions 62 and 73 of the BRODA sequence, the first 13 direction numbers used to generate the first 2¹³ points have the same MSB. Similarly, the first 10 direction numbers of the Joe/Kuo Sobol' sequence share the same MSB across dimensions 358 and 401. Note that Figure 15a is an example where the projection of the first 8 direction numbers of the sequence across dimensions 47 and 49 share the same second MSB.

Acknowledgments

The authors would like to thank Alex Tikhonov for many useful discussions and the implementation of these methods in our software suite, Marco Bianchetti, Olivier Guiavarc'h, Sergei Kucherenko, Leslie Ng, Christoph Puetter, and Stefano Scoleri for their helpful comments, Andy Aziz, Allan Cowan, and Yaacov Mutnikas for their support and encouragement to do this work.

References

- Andersen, L., Pykhtin, M., and Sokol, A. 2017. Rethinking margin period of risk. Journal of Credit Risk 13(1), 1–45.
- [2] Bianchetti, M., Kucherenko, S., and Scoleri, S. 2015. Pricing and risk management with high-dimensional quasi-Monte Carlo and global sensitivity analysis. *Wilmott Magazine* 79, 46–70.
- [3] Bianchetti, M., Kucherenko, S., and Scoleri, S. 2016. High-dimensional quasi-Monte Carlo methods: multi-asset options and FD vs AAD Greeks [Power-Point presentation]. Available at: https://www.slideshare.net/marcobianchetti/ high-dimensional-quasi-monte-carlo-method-in-finance (Accessed: 30 Sept 2019).

- [4] http://www.broda.co.uk/
- [5] Burnett, B., O'Callaghan, S., and Hulme, T. 2016. Risk optimisation: the noise is the signal. *Risk* August, 67–71.
- [6] Caflisch, R.E., Morokoff, W., and Owen A. 1997. Valuation of mortgage backed securities using Brownian bridges to reduce effective dimensions. *Journal of Computational Finance* 1(1), 27-46.
- [7] Capriotti, L., Lee, J., and Peacock, M. 2011. Real time counterparty credit risk management in Monte Carlo. *Risk* 24(6), 82–86.
- [8] Ghamami, S., and Zhang, B. 2014. Efficient Monte Carlo counterparty credit risk pricing and measurement. *Journal of Credit Risk* 10(3), 87–133.
- [9] Giles, M., and Glasserman, P. 2006. Smoking Adjoints: fast evaluation of Greeks in Monte Carlo calculations. *Risk* 19(1), 88–92.
- [10] Glasserman, P. 2004. Monte Carlo Methods in Financial Engineering, Springer.
- [11] Gregory, J. 2015. The xVA Challenge: Counterparty Credit Risk, Funding, Collateral and Capital, Wiley.
- [12] Hull, J. and White, A.D. 1994. Numerical procedures for implementing term structure models. I. Single-factor models. *The Journal of Derivatives* 2(1), 7–16.
- [13] Jäckel, P. 2002. Monte Carlo Methods in Finance, Wiley Finance.
- [14] http://web.maths.unsw.edu.au/~fkuo/sobol/new-joe-kuo-6.21201
- [15] Joe, S., Kuo, FY. 2008. Constructing Sobol' sequences with better two-dimensional projections. SIAM Journal on Scientific Computing 30(5), 2635–2654.
- [16] Kucherenko, S., Shah, N. 2007. The Importance of being Global. Application of Global Sensitivity Analysis in Monte Carlo option Pricing. *Wilmott Magazine* 4, 82–91.
- [17] Kucherenko, S., Feil B., Shah, N., Mauntz, W. 2011. The identification of model effective dimensions using global sensitivity analysis. *Reliability Engineering and System Safety* 96(4), 440–449.
- [18] L'Ecuyer, P. 2009. Quasi-Monte Carlo methods with applications in finance, Finance and Stochastics 13(3), 307–349.
- [19] Lin, J., Wang., X. (2008). New Brownian bridge construction in quasi-Monte Carlo methods for computational finance. *Journal of Complexity* 24(2) 109–133.
- [20] Matoušek, J. (1998). On the L_2 -discrepancy for anchored boxes. Journal of Complexity 14(4), 527–556.
- [21] Matsumoto, M., and Nishimura, T. 1998. Mersenne Twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *Transactions on Modeling and Computer Simulation* 8(1), 3–30.
- [22] Morokoff, W., Caflisch, R. 1995. Quasi-Monte Carlo integration. Journal of Computational Physics 122(2), 218–230.
- [23] Ng, L., Peterson, D., and Rodriguez, A. 2010. Potential future exposure calculations of multi-asset exotic products using the stochastic mesh method. *Journal of Computational Finance* 14(2), 119–153.

- [24] Niederreiter, H. 1992. Random Number Generation and Quasi-Monte Carlo Methods. Philadelphia: Society for Industrial and Applied Mathematics.
- [25] Owen, A.B. 1997. Scrambled net variance for integrals of smooth functions. Annals of Statistics 25(4), 1541–1562.
- [26] Owen, A.B. 1998. Scrambling Sobol' and Niederreiter-Xing points. Journal of Complexity 14(4), 466–489.
- [27] Pykhtin, M. 2009. Modeling credit exposure for collateralized counterparties, The Journal of Credit Risk 5(4), 3–27.
- [28] Sobol', I. 1967. On the Distribution of points in a cube and the approximate evaluation of integrals. USSR Journal of Computational Mathematics and Mathematical Physics 7(4), 86–112.
- [29] Sobol', I., Asotsky, D., Kreinin, A., Kucherenko, S. 2011. Construction and comparison of High-Dimensional Sobol generators. *Wilmott Magazine* 56, 64–79.
- [30] Tan, K. and Boyle, P. 2000. Applications of randomized low discrepancy sequences to the valuation of complex securities. *Journal of Economic Dynamics and Control* 24(11), 1747–1782.
- [31] Wang, X. and Sloan, I.H. 2001. Quasi-Monte Carlo methods in financial engineering: an equivalence principle and dimension reduction. *Journal of Operations Research* 59(1), 80–95.