Accelerating CVA and CVA Sensitivities using Quasi Monte Carlo Methods

Stefano Renzitti

IHS Markit, 1066 West Hastings Street, Vancouver, BC V6E 3X1, Canada
stefano.renzitti@ihsmarkit.com

June 8, 2018

Abstract

We analyze the efficiency of the quasi Monte Carlo method (QMC) when used to compute credit valuation adjustment (CVA) and CVA sensitivities for various portfolios of interest rate swaps using a multi-currency, multi-curve extension to the Hull-White model. We find that QMC with Sobol sequences (using Broda's SobolSeq65536 direction numbers) and the Brownian bridge discretization produces results as accurate as classical MC with 10,000 simulations when using, on average, 796 simulations, a factor of 13 acceleration. The acceleration varies significantly across portfolios (increasing with moneyness and usually, but not always, decreasing with number of factors), calculation types (order from highest to lowest, usually, but not always, CVA, CR Deltas, IR and FX Deltas, and IR and FX Vegas), and the choice of model (local models usually outperform global models). QMC without the Brownian bridge discretization results in a more modest acceleration of 4. Classical MC with antithetic sampling results in a factor of 2 acceleration. Randomized QMC with the Brownian bridge discretization resulted in a factor of 6 acceleration in the cases tested. Randomization, in combination with a direct and independent simulation approach of the risk factors, produced extraordinary results in the limited tests we performed (CVA on a single currency swap), offering an acceleration of roughly 300 times.

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

Contents

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 payoff in this case is the netted portfolio value less collateral (floored at zero) at the time of counterparty default, multiplied by one minus the recovery rate:

$$\mathrm{CVA}_t = \mathbb{E}_t\left[\int_t^T (1-R_c)\max(V_u - C_u, 0) D_{t,u}\lambda_{c,u} e^{-\int_t^u(\lambda_{c,s}+\lambda_{b,s})ds} du\right], \quad (1)$$

where V_u is the netted idealized value of all trades held against the counterparty c at time u, C_u is the collateral posted by the counterparty at time u, R_c is the recovery rate if the counterparty defaults, $D_{t,u}$ is the stochastic discount factor associated with the risk adjusted probability measure used in the expectation, λ_c is the default intensity of the counterparty (we have assumed a reduced form model), λ_b is the default intensity of the bank computing CVA, and T is the maturity date of the portfolio. See Gregory 2015 [11] for more details. The payoff is at counterparty level, potentially path dependent (collateral, early exercise conditions, lags between fixings and cash flows), and subject to change. The expectation of high dimensional, fluid payoffs of this sort are in practice estimated with Monte Carlo (MC) simulation [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 excellent thorough explanations of Monte-Carlo 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 20,000,000,000,000 trade price evaluations per day.

Not surprisingly, quants have been searching for ways to accelerate this massive calculation. One successful line of research uses algorithmic adjoint 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 [8] and Capriotti et al. 2011 [6] for more information). Assuming a conservative fixed multiple of 10, this would reduce the total number of calculations by a factor of 20, requiring 1,000,000,000,000 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 [7] highlight that direct and independent simulation of the portfolio value to each time step, rather than the usual chronological and sequential 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 portfolios containing collateral may not benefit as a result, but the technique looks quite promising for portfolios of uncollateralized vanillas.

In a similar line of research, Burnett, O'Callaghan and Hulme 2016 [4] 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 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 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 don't 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, Morokov, and Owen 1997 [5]. 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. 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 for 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 (non collateralised vanilla interest rate swap portfolios) and a specific model (multi-currency, multi-curve extension to the Hull-White model [12] with deterministic hazard rates).

2 Portfolios, Model and Calculations

We create portfolios with one, six, and eleven 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 300bps), in the money (fixed rates set to par minus 100bps), at the money, out of the money (fixed rate set to par plus 100bps), to far out of the money (fixed rate set to par plus 300bps). 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 rate simulated by the model. A cross currency, multi-curve extension to the Hull-White model is used to simulate these quantities [12]. One Hull-White factor is used to simulate interest rates in each currency (the spread between interest rates in the same currency is assumed to be deterministic). Each exchange rate is log-normally distributed. Credit events are assumed to be independent of the exposure profiles and numeraire, and thus do not directly affect the simulation. In total, the single currency portfolio depends on one factor, the six currency portfolio depends on 10 factors, and the 11 currency portfolio depends on 20 factors. Exposure dates are quarterly spaced, with additional exposures added the day after the underlying swap's cash flows

(61 in total).

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 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 options. Correlations are estimated historically.

We compute CVA and CVA sensitivities to parallel shifts of the curves. For each currency (except the domestic), we compute four sensitivities: IR Delta (a one basis point shift to each point of the yield curve), IR Vega (a one percent shift to each point of the swaption volatility matrix), FX Delta (a one tenth of a percent shift to the spot FX rate), and FX Vega (a one percent shift to each point of 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 one currency domestic portfolios require three sensitivities (one CR Delta, one IR Delta, and one IR Vega), the six currency portfolios require 23 sensitivities (one CR Delta, six IR Deltas, six IR Vegas, five FX Deltas, and five FX Vegas), and the 11 currency portfolios require 43 sensitivities (one 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.

A typical multi-step Monte-Carlo simulation is used to estimate CVA and CVA sensitivities, including a simple Euler scheme to evolve the risk factors (natural log of the FX rate). Details are provided in appendix A. Four Monte Carlo variations are used:

Classical Monte Carlo (MC): Mersenne Twister pseudo random numbers [14] are used to generate n vectors of uniform random variables. The error is normally distributed with standard deviation equal to:

$$\sigma_{\mathtt{MC}} = \sigma_{\mathtt{Payoff}} n^{-\frac{1}{2}} \tag{2}$$

where $\sigma_{\mathtt{Payoff}}$ is the standard deviation of the payoff and is estimated as the sample standard deviation of the payoff.

Classical Monte Carlo with Antithetic Sampling (AMC): Antithetic sampling attempts to reduce the variance of the Monte Carlo estimator by introducing negative dependence between pairs of random replications. Assume n is an even number. The error is normally distributed with standard deviation equal to:

$$\sigma_{\texttt{AMC}} = \sqrt{2}\sigma_{\texttt{Pairs}} n^{-\frac{1}{2}} \quad (3)$$

where $\sigma_{\texttt{Pairs}}$ is the standard deviation of the $n/2$ samples computed by first averaging the antithetic pairs. The convergence rate is the same as classical MC, but the constant is smaller if antithetic pairs are negatively correlated.

Quasi-Monte Carlo Broda (QMC): Of the low discrepancy sequences available, Sobol sequences [16] have been shown to have superior convergence properties, to be the most robust to the number of dimensions, and very efficient to generate (see Glasserman 2204 [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 commercially that allow for generation of 65,536 dimensions [3]. The direction numbers, constructed in conjunction with Ilya Sobol, were chosen such that all 65K dimensions satisfy property A, and that all adjacent five dimensions satisfy property 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 firsts 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 split them equally into four. Glasserman states, "A general strategy for improving QMC approximations applies change of variables to produce an integrand for which only a small number of arguments are "important" and then applies the lowest-indexed coordinates of a QMC sequence to those coordinates."[10]. We use one such method here, and another in the next method described below. Correlated Brownian increments are computed as the weighted sum of the independent normal random variables. The weights are equal to the square root of the correlation matrix, computed using an Eigen decomposition, ordered from largest to smallest Eigen values. This increases the payoff's dependence on fewer and earlier indexed uniform random variables. There is no known statistical distribution for the error. The standard deviation (defined as the root mean squared error RMSE in this paper), can be estimated empirically and assuming it follows a power law, it is shown in the appendix B to be equal to:

$$\sigma_{\texttt{QMC}} = \sigma_{\texttt{Payoff}} n^{-\beta(\texttt{Payoff},\texttt{QMC})} \quad (4)$$

where the convergence rate β is usually between 1/2 and 1 depending on the payoff and risk factor simulation.

Quasi-Monte Carlo Broda + Brownian Bridge (QMC + BB): Non-collateralized non-path dependent netting sets exposures primarily depend on the risk factor values at the corresponding exposure dates. The *Brownian bridge (BB) discretization* [5] can be used to capture the primary features of these risk factor paths with typically far fewer random numbers than the traditional chronological simulation approach. More specifically, we use the Brownian bridge discretization to map the independent normal random variables into independent Brownian motions, one for each factor. Correlated Brownian increments are then computed as the weighted sum of the independent Brownian increments, the weights set equal to the square root of the correlation matrix, computed using an Eigen decomposition, ordered from largest to smallest Eigen values. Under assumptions described in appendix B, the standard deviation, or root mean squared error, can be written as:

$$\sigma_{\texttt{QMCBB}} = \sigma_{\texttt{Payoff}} n^{-\beta(\texttt{Payoff},\texttt{QMCBB})} \tag{5}$$

All simulations for the base and bumped runs are computed using the same random numbers. The correlations (and thus Eigen decomposition) are held constant between bumps, ensuring random numbers are interpreted in the same way. Failure to do so produces unusable results, as the standard error of the sensitivities approaches the standard error of the level, which is typically much bigger than the sensitivity being measured. See table 7, 8, and 10.

3 Results

For each of the portfolios, calculation types, and the four Monte Carlo variants described, we estimate the convergence coefficients/constants empirically as described in appendix B. We then find the expected number of paths needed to match the expected error of classical MC when using 10,000 paths.

Results for classical MC with antithetic paths are presented first in table 1. On average, 5,759 antithetic paths are needed in order to match the expected error of 10,000 classical MC paths, roughly a factor of two acceleration. The acceleration varies significantly with moneyness, consistently increasing as the portfolio becomes more in the money. Antithetic sampling works best when the payoff becomes more linear and the antithetic samples become more negatively correlated. The results for the far out of the money portfolio are slightly worse than classical MC. The acceleration also varies with the calculation complexity, most notably offering smaller gains for Vegas. Portfolio size does not appear to affect the efficiency[1].

Quasi random numbers are even more effective, requiring only 2,803 paths to produce expected errors roughly equivalent in size to 10,000 classical MC paths: a factor of four acceleration. The number of paths needed to match classical MC with 10,000 paths varies significantly between portfolios and calculation type,

[1]When there would be more than one result for a particular entry in the matrix, such as IR and FX Deltas and Vegas for the six and eleven currency portfolios, we aggregate by averaging the results. Consequently, the displayed values do not provide the equivalent paths for the error of adding up all CVA and CVA sensitivities for all portfolios.

Table 1: Approximate number of antithetic paths needed to produce CVA and CVA sensitivities with errors roughly equivalent to classical MC with 10,000 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread). *Local models* are used for each portfolio. Details of equivalent path methodology can be found in appendix B.

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	285	715	811	604
-300bps	CR Delta	227	546	613	462
-300bps	IR, FX Delta	1,473	1,682	1,727	1,627
-300bps	IR, FX Vega	2,980	3,071	3,377	3,143
-300bps	Average	1,241	1,504	1,632	1,459
-100bps	CVA	2,025	1,576	1,833	1,811
-100bps	CR Delta	1,699	1,225	1,449	1,458
-100bps	IR, FX Delta	1,526	2,455	2,396	2,126
-100bps	IR, FX Vega	8,171	6,271	5,844	6,762
-100bps	Average	3,355	2,882	2,880	3,039
0bps	CVA	5,099	5,452	4,595	5,048
0bps	CR Delta	4,960	5,346	4,314	4,873
0bps	IR, FX Delta	1,953	4,938	4,616	3,835
0bps	IR, FX Vega	7,560	8,925	8,442	8,309
0bps	Average	4,893	6,165	5,492	5,517
100bps	CVA	8,966	9,323	7,345	8,545
100bps	CR Delta	8,731	9,046	7,391	8,389
100bps	IR, FX Delta	5,202	9,583	7,586	7,457
100bps	IR, FX Vega	8,136	8,713	8,857	8,569
100bps	Average	7,759	9,167	7,795	8,240
300bps	CVA	10,815	7,450	10,195	9,487
300bps	CR Delta	10,710	7,267	10,320	9,432
300bps	IR, FX Delta	12,172	9,922	10,922	11,005
300bps	IR, FX Vega	12,154	11,962	12,570	12,228
300bps	Average	11,462	9,150	11,002	10,538
	Average	5,742	5,773	5,760	5,759

however, increasing as the portfolio becomes more out of the money (887 paths for far in the money, 1,309 paths for in the money, 1,902 paths for at the money, 3,051 paths for out of the money, and 6,864 paths for far out of the money); usually, but not always, increasing as more factors are added to the portfolio (1,499 paths for the one currency one swap portfolios and one currency local model, 3,489 paths for the six currency six swap portfolios and six currency local model, and 3,419 paths for the eleven currency eleven swap portfolios and eleven currency local model); and usually, but not always, increasing with the calculation complexity (2,238 paths for CVA, 2,195 paths for CR Delta, 2,736 paths for IR and FX Deltas, and 4,041 paths for IR and FX Vegas). Illustrative results are presented in table 2. The clear improvement in QMC efficiency when the portfolio becomes more in the money is most likely caused by the payoff becoming more separable as flooring the portfolio at zero rarely happens (reducing the effective dimension). Far out of the money portfolios are the most difficult but also result in the smallest CVA and sensitivities (see table 7).

Quasi random numbers with the Brownian bridge discretization perform the best, requiring only 796 paths to produce results as accurate as classical MC with 10,000, a factor of 13 acceleration. Similar to QMC, the performance varies significantly between portfolios and calculation type, increasing as the portfolio becomes more out of the money (320 paths for far in the money, 428 paths for in the money, 620 paths for at the money, 855 paths for out of the money, and 1,755 paths for far out of the money); usually, but not always, increasing as more factors are added to the portfolio (264 paths for the one currency one swap portfolios and one currency local model, 923 paths for the six currency six swap portfolios and six currency local model, and 1,201 paths for the eleven currency eleven swap portfolios and eleven currency local model); and usually, but not always, increasing with the calculation complexity (521 paths for CVA, 538 paths for CR Delta, 829 paths for IR and FX Deltas, and 1,295 paths for IR and FX Vegas). Illustrative results are presented in table 3.

Additional tables, including the root mean squared error (RMSE), convergence coefficients β, and convergence constants for the various methods can be found in appendix C.

4 Global versus Local 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 modelling all portfolios, to simulate each counterparty level portfolio rather than a *local model*, capable of modelling only the risk factors needed for that counterparty. This is inconsequential for classical MC, as the convergence rate is fixed at 1/2, but can have a real impact on the QMC performance, as illustrated in table 4. The table contains CVA and CVA sensitivity estimates for a single USD swap using global models capable of modelling one,

Table 2: Approximate number of QMC paths needed to produce CVA and CVA sensitivities with errors roughly equivalent to classical MC with 10,000 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread). *Local models* are used for each portfolio. Details of equivalent path methodology can be found in appendix B.

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	311	724	627	554
-300bps	CR Delta	281	587	535	467
-300bps	IR, FX Delta	483	1,042	1,290	939
-300bps	IR, FX Vega	853	1,978	1,934	1,588
-300bps	Average	482	1,083	1,096	887
-100bps	CVA	519	1,009	820	783
-100bps	CR Delta	460	803	686	650
-100bps	IR, FX Delta	428	1,529	1,647	1,201
-100bps	IR, FX Vega	957	3,699	3,146	2,601
-100bps	Average	591	1,760	1,575	1,309
0bps	CVA	653	1,572	1,118	1,114
0bps	CR Delta	567	1,286	891	915
0bps	IR, FX Delta	708	2,539	2,373	1,874
0bps	IR, FX Vega	876	5,459	4,778	3,704
0bps	Average	701	2,714	2,290	1,902
100bps	CVA	1,415	3,474	2,295	2,395
100bps	CR Delta	1,417	3,478	2,204	2,366
100bps	IR, FX Delta	1,009	4,551	3,577	3,046
100bps	IR, FX Vega	1,010	6,266	5,918	4,398
100bps	Average	1,213	4,442	3,499	3,051
300bps	CVA	4,847	6,311	7,876	6,345
300bps	CR Delta	5,166	6,534	8,034	6,578
300bps	IR, FX Delta	3,803	7,881	8,177	6,620
300bps	IR, FX Vega	4,224	9,052	10,460	7,912
300bps	Average	4,510	7,455	8,637	6,864
	Average	1,499	3,489	3,419	2,803

Table 3: Approximate number of QMC + BB paths needed to produce CVA and CVA sensitivities with errors roughly equivalent to classical MC with 10,000 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread). *Local models* are used for each portfolio. Details of equivalent path methodology can be found in appendix B.

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	164	221	369	251
-300bps	CR Delta	170	224	311	235
-300bps	IR, FX Delta	195	301	480	325
-300bps	IR, FX Vega	265	373	772	470
-300bps	Average	199	280	483	320
-100bps	CVA	174	269	296	246
-100bps	CR Delta	179	270	279	243
-100bps	IR, FX Delta	185	552	664	467
-100bps	IR, FX Vega	253	740	1,279	757
-100bps	Average	198	458	630	428
0bps	CVA	193	379	533	368
0bps	CR Delta	201	428	488	372
0bps	IR, FX Delta	237	751	872	620
0bps	IR, FX Vega	228	1,172	1,962	1,121
0bps	Average	215	683	964	620
100bps	CVA	240	476	813	510
100bps	CR Delta	262	552	761	525
100bps	IR, FX Delta	304	1,316	1,225	948
100bps	IR, FX Vega	238	1,686	2,386	1,437
100bps	Average	261	1,008	1,296	855
300bps	CVA	383	1,514	1,796	1,231
300bps	CR Delta	427	1,602	1,916	1,315
300bps	IR, FX Delta	483	2,218	2,647	1,783
300bps	IR, FX Vega	487	3,422	4,166	2,691
300bps	Average	445	2,189	2,631	1,755
	Average	264	923	1,201	796

six, eleven, and thirty currencies. The results deteriorate from the one currency one factor *local model* case to the global models with various number of currencies (erratically) but are nevertheless still significantly better than classical MC, even at the eleven and thirty currency mark. By comparing table 3 and 4 we see that the deterioration in performance, as a function of the number of currencies, is roughly the same even though the payoff of the latter does not change, only more uniform random variables are used to describe the same USD risk factors.

5 Error Estimates using Randomization

Because of the large variation in the efficiency of QMC across portfolios, calculation types, and simulation models, it is important for anyone using this technique in practice to be able to conveniently evaluate the efficiency on the portfolios and calculations of most concern to them. Estimating the efficiency requires an estimate of the QMC and MC errors. MC errors can be computed easily enough, but QMC error estimates are more difficult, requiring either some modelling assumptions and multiple runs using various segments of the sequence (described in appendix B), or randomization techniques, such as *digital shift* (see Giles et al. 2008 [9] for an excellent summary). Randomization produces a valid standard error and confidence intervals, but the user must make a trade off between the accuracy of the error estimate and the accuracy of the answer itself using the number of randomized trials as a dial. Under assumptions described in appendix B, the randomized quasi Monte Carlo root mean squared error for a payoff f grows by the factor $k^{\beta-1/2}$ over the regular QMC estimate, where k is the number of randomized trials and β is the payoff specific convergence rate of the QMC estimate, typically between 1/2 and 1 depending on the effective dimension. Giles et al. 2008 [9] suggest using between 10 and 20 randomized trials as a compromise: we use 16. The results are presented in table 5. The average number of paths taken to match 10,000 classical MC paths increases to 1,589, a more modest factor of 6 acceleration. Reducing the number of randomized trials to 8, 4, and 2 results in accelerations of roughly 7, 9, and 10 times, respectively.

6 Direct Simulation with Randomization

The various acceleration approaches presented above are not mutually exclusive: they can be used together to compound the computational savings. One potentially interesting combination is to combine the direct simulation methods proposed by Ghamami and Zhang 2014 [7] for non path dependent CVA calculations with QMC and the Brownian Bridge mechanism, allowing us to first increase the payoff's dependence onto fewer, earlier indexed uniform random

Table 4: Approximate number of QMC + BB paths needed to produce CVA and CVA sensitivities with errors roughly equivalent to classical MC with 10,000 paths for a single USD swap with various fixed swap rates (set to par plus the given spread) and global models (modelling in total one, six, eleven, and thirty currencies). Details of equivalent path methodology can be found in appendix B.

Spread	Type	1 CCY	6 CCY	11 CCY	30 CCY	Avg.
-300bps	CVA	164	215	367	258	280
-300bps	CR Delta	170	199	291	266	252
-300bps	IR, FX Delta	195	299	634	398	444
-300bps	IR, FX Vega	265	492	1,115	600	736
-300bps	Average	199	301	602	380	428
-100bps	CVA	174	346	724	448	506
-100bps	CR Delta	179	325	588	520	478
-100bps	IR, FX Delta	185	303	527	412	414
-100bps	IR, FX Vega	253	773	1,931	973	1,226
-100bps	Average	198	437	942	588	656
0bps	CVA	193	422	1,208	626	752
0bps	CR Delta	201	487	1,141	859	829
0bps	IR, FX Delta	237	556	760	567	628
0bps	IR, FX Vega	228	546	1,526	897	990
0bps	Average	215	503	1,159	737	800
100bps	CVA	240	636	1,705	913	1,085
100bps	CR Delta	262	713	1,710	1,397	1,273
100bps	IR, FX Delta	304	895	1,556	939	1,130
100bps	IR, FX Vega	238	653	1,660	1,002	1,105
100bps	Average	261	724	1,658	1,063	1,148
300bps	CVA	383	1,491	3,772	1,706	2,323
300bps	CR Delta	427	1,488	3,593	2,097	2,393
300bps	IR, FX Delta	483	1,472	3,269	1,909	2,217
300bps	IR, FX Vega	487	1,358	3,339	1,700	2,132
300bps	Average	445	1,452	3,493	1,853	2,266
	Average	263	683	1,571	924	1,060

Table 5: Approximate number of RQMC + BB ($k = 16$ trials) paths needed to produce CVA and CVA sensitivities with errors roughly equivalent to classical MC with 10,000 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread). Details of equivalent path methodology can be found in appendix B.

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Avg.**
-300bps	CVA	592	816	797	735
-300bps	CR Delta	612	806	989	736
-300bps	IR, FX Delta	635	906	1,066	869
-300bps	IR, FX Vega	714	1,073	1,300	1,029
-300bps	Average	638	900	988	842
-100bps	CVA	620	850	884	785
-100bps	CR Delta	631	837	863	777
-100bps	IR, FX Delta	625	1,144	1,319	1,030
-100bps	IR, FX Vega	798	1,534	1,921	1,418
-100bps	Average	669	1,091	1,247	1,002
0bps	CVA	691	939	1,068	900
0bps	CR Delta	719	965	1,037	907
0bps	IR, FX Delta	675	1,345	1,734	1,251
0bps	IR, FX Vega	775	2,282	2,816	1,958
0bps	Average	715	1,383	1,664	1,254
100bps	CVA	852	1,247	1,318	1,139
100bps	CR Delta	900	1,331	1,373	1,201
100bps	IR, FX Delta	783	2,042	2,259	1,695
100bps	IR, FX Vega	815	3,149	3,681	2,548
100bps	Average	837	1,942	2,158	1,646
300bps	CVA	1,303	3,146	3,460	2,636
300bps	CR Delta	1,393	3,264	3,727	2,795
300bps	IR, FX Delta	1,368	3,516	4,099	2,994
300bps	IR, FX Vega	1,399	5,690	6,028	4,372
300bps	Average	1,366	3,904	4,328	3,199
	Average	845	1,844	2,077	1,589

Table 6: CVA error for an at the money (ATM) 10 year USD fixed for float swap with $10,000 notional for various numbers of simulation paths. Three methodologies are presented. Classical MC, randomized QMC + BB (RQMCBB) with regular pathwise simulation, and randomized QMC + BB (RQMCBB) with direct and independent simulation of the risk factors to each time step.

Paths	**Pseudo Pathwise**	**RQMCBB Pathwise**	**RQMCBB Direct**
32	2.7898	0.7904	0.1957
64	1.9505	0.4156	0.0759
128	1.6297	0.2383	0.0440
256	1.0343	0.1347	0.0247
512	0.7116	0.0695	0.0133
1,024	0.5221	0.0328	0.0059
2,048	0.3384	0.0176	0.0032
4,096	0.2281	0.0109	0.0017
8,192	0.1716	0.0064	0.0010
16,384	0.1248	0.0038	0.0006
32,768	0.0992	0.0020	0.0003

variables, and second, when combined with a randomization method such as *digital shift* [10], where we only use one randomized trial per exposure date, diversify the errors across the time axis. Preliminary convergence results are provided in table 6. Remarkably, roughly only 32 RQMC+BB paths are needed to match accuracy of 10,000 classical Monte-Carlo paths with pathwise simulation.

7 Conclusion

QMC with the Brownian bridge discretization and an Eigen decomposition works very well on portfolios of non-collateralized swaps, offering roughly a factor of 13 acceleration over the classical MC counterpart in the cases tested. The size of the improvement varies significantly between portfolios, calculation types, and simulation models. Obtaining the optimal acceleration requires reformulating the payoff to depend on fewer and earlier indexed elements of the quasi uniform variables [10]. Brownian bridge, Eigen decomposition, local simulation models, and direct simulation are all examples of this.

The success of these methods depends on the payoff. The Brownian bridge discretization will most likely become less effective when used with collateralized portfolios with small thresholds where the margin period of risk dominates the exposure, for example, as it will depend more on the difference between the portfolio values at adjacent exposure dates rather than the level. Furthermore, portfolios with more exposure up front may not perform as well with QMC and the Brownian bridge mechanism, as more and later indexed random variables would be used to explain that part of the path.

Nevertheless, as Jackel states: "The lesson to learn with respect to well initialised Sobol' numbers is that they will provide a substantial performance boost in lower dimensions, and will still work at least as well as pseudo random number generators in higher dimensions."[13]. The primary reason not to use QMC, aside from the cost of developing it, is then the missing error estimate. Randomized QMC provides a nice compromise, producing errors between those obtained by classical MC and QMC, where the number of randomized trials k can be used to adjust the error between these two extremes. Setting the number of randomized trials to 16 in our tests resulted in a more modest but still impressive acceleration factor of 6.

Randomization, in combination with a direct and independent simulation approach of the risk factors, produced extraordinary results in the limited test we performed (CVA on a single currency swap), offering an acceleration of roughly 300 times. In the future, we would like to test this on larger portfolios and CVA sensitivities.

Acknowledgements

I would like to thank Pouya Bastani, Steven Sivorot, and 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

[1] Bianchetti M, Kucherenko S, Scoleri S. Pricing and Risk Management with High-Dimensional Quasi-Monte Carlo and Global Sensitivity Analysis. *Wilmott* July:46–70 (2015)

[2] Bianchetti M, Kucherenko S, Scoleri S., Application of Quasi Monte Carlo and Global Sensitivity Analysis to Option Pricing and Greeks (February 5, 2017). Available at SSRN: https://ssrn.com/abstract=2911698 or http://dx.doi.org/10.2139/ssrn.2911698

[3] `http://www.broda.co.uk/`

[4] Burnett, Benedict and O'Callaghan, Simon C and Hulme, Tom, Risk Optimisation: Finding the Signal in the Noise (May 26, 2016). Available at SSRN: https://ssrn.com/abstract=2784702 or http://dx.doi.org/10.2139/ssrn.2784702

[5] Caflisch RE, Morokoff WJ, Owen A.B. Valuation of mortgage backed securities using Brownian bridges to reduce effective dimensions. *Journal of Computational Finance*, 1(1):27–46 (1997)

[6] L. Capriotti, J. Lee, and M. Peacock. Real Time Counterparty Credit Risk Management in Monte Carlo. *Risk*, pages 82-86, (May 2011). Available at http://papers.ssrn.com/abstract=1824864.

[7] Ghamami, S., Zhang, B. Efficient Monte Carlo counterparty credit risk pricing and measurement. *Journal of Credit Risk*, Volume 10, Number 3:87–133 (Sept 2014)

[8] Giles, M. and Glasserman, P.. Smoking adjoints: fast Monte Carlo Greeks. Risk, pages 92-96 (January 2006).

[9] Giles, M., Kuo, F., Sloan, I., Waterhouse, B. Quasi Monte Carlo for finance applications. *ANZIAM*, Volume 50:c308–c323 (Nov 2008)

[10] Glasserman P. Monte Carlo Methods in Financial Engineering, New York: Springer-Verlag, ISBN 0-387-00451-3 (2004)

[11] Gregory J. The xVA Challenge: Counterparty Credit Risk, Funding, Collateral and Capital, Wiley, ISBN 978-1-119-10941-9 (2015)

[12] Hull J, White A. Numerical procedures for implementing term structure models. I. Single-factor models. *Journal of Derivatives*,2:37–48 (1994)

[13] Jäckel P. Monte Carlo methods in finance, Wiley Finance, ISBN 0-471-49741-X (2002)

[14] Matsumoto, M., Nishimura, T. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation.* 8 (1): 3-30. (1997)

[15] Morokoff WJ, Caflisch RE. Quasi-Monte Carlo Integration. *Journal of Computational Physics*, 122:218–230 (1995)

[16] Sobol' IM. On the distribution of points in a cube and the approximate evaluation of integrals, *USSR Journal of Computational Mathematics and Mathematical Physics*, 16:1332–1337 (1967)

[17] Sobol IM, Kucherenko S. On global sensitivity analysis of quasi-Monte Carlo algorithms. *Monte Carlo Methods and Appl*, Vol. 11, No. 1: 1–9 (2005).

[18] Sobol' IM, Asotsky D, Kreinin A, Kucherenko S. Construction and Comparison of High-Dimensional Sobol Generators, *Wilmott Journal*, Nov:64–79 (2012).

Appendices

A Monte Carlo CVA Estimate

We choose a time discretization $t^e = \{t_0^e, t_1^e, t_2^e, \ldots, t_{N-1}^e\}$, with $t_0^e = 0 < t_1^e < t_2^e < \ldots < t_{N-1}^e \leq T$ and label them exposure dates. We further assume a reduced form credit model setup with default times independent from each other and the exposures, and we get an approximation to CVA:

$$\hat{\mathrm{CVA}}_t = \mathbb{E}_t \left[\sum_{i=1}^{N-1} (1 - R_c) \max(V_{t_i^e} - C_{t_i^e}, 0) D_{t,t_i^e} (S^c_{t,t_{i-1}^e} - S^c_{t,t_i^e}) S^b_{t,t_i^e} \right] \tag{6}$$

where $S^a_{t,T}$ is the probability party a survives up until T, as seen from time t. We consider a financial model that is Markovian in a vector of n_{factor} risk factors X_t, where the evolution of X_t is goverened by a diffusion process:

$$dX_t = [AX_t + f_t]\, dt + \Sigma_t dW \tag{7}$$

where A is a matrix of constants, f_t is a deterministic vector function of time, Σ_t is a matrix of time dependent coefficients, and dW is a vector of n_{factor} correlated Brownian motions, with instantaneous correlations equal to $R_{i,j}$ for $i, j = 0, 1, \ldots, n_{factor} - 1$. The cross currency multi-curve Hull White model described above fits into this class of models, and is what we use throughout this paper.

The risk factor vector X_t is required on the union of the exposure dates and portfolio fixing dates. Define the total set of fixing dates as $t^f = \{t_0^f, t_1^f, t_2^f, \ldots, t^f_{n^{\text{fixing}}}\}$, where $t_0^f = 0 < t_1^f < t_2^f < \ldots < t^f_{n^{\text{fixing}}} \leq T$ and $t^e \in t^f$. Using a basic Euler scheme, we explicitly write the distribution of risk factors in terms of uniform random variables, independent Wiener processes, and correlated Wiener processes. Starting at $t_0^f = 0$ and X_0 given, chronologically iterate over the fixing indices $i = 1, 2, \ldots, n^{\text{fixing}}$:

$$\xi_i = \Phi^{-1}(\mathrm{u}_i) \tag{8}$$

$$\hat{W}_{t_i^f} = \hat{W}_{t_{i-1}^f} + \xi_i \sqrt{\Delta_i} \tag{9}$$

$$W_{t_i^f} = W_{t_{i-1}^f} + \sqrt{R}\left[\hat{W}_{t_i^f} - \hat{W}_{t_{i-1}^f}\right] \tag{10}$$

$$X_{t_i^f} = X_{t_{i-1}^f} + \left[AX_{t_{i-1}^f} + f_{t_{i-1}^f}\right]\Delta_i + \Sigma_{t_{i-1}^f}\left[W_{t_i^f} - W_{t_{i-1}^f}\right] \tag{11}$$

where u_i is an n^{factor} vector of uniformly distributed random variables, $\Phi^{-1}(x)$ is the inverse cumulative normal function operating element wise on x, ξ_i is an n^{factor} vector of standard normal random variables, $\hat{W}_t$ is an n^{factor} vector of independent Wiener processes at time t with $\hat{W}_0 = \mathbf{0}$, $\mathbf{0}$ is an n^{factor} vector of zeros, $\Delta_i = t_i^f - t_{i-1}^f$, W_t is an n^{factor} vector of correlated Wiener processes as defined above, and $\sqrt{R}$ is the n^{factor} by n^{factor} matrix containing the square root correlation matrix in the Eigen decomposition sense. The square root of a matrix is not unique. Another common method is to use a Cholesky decomposition. We chose the Eigen decomposition over the Cholesky decomposition as it sorts the factors according to the total explained variation, something particularly useful for QMC.

The portfolio idealized value V_u is a function of the current model state X_u and can be a function of past model states $X_{s<u}$ too, the dependence a result of future portfolio cashflows depending on past fixings, likewise with the collateral balance C_u and stochastic discount factor $D_{t,u}$. The CVA payoff is typically computed in step with the risk factors, evolving risk factors to each fixing date, recording fixings needed by portfolio and or collateral, then evolving the risk factors to the next exposure date, evaluating the portfolios and collateral balances, and finally the CVA term for the current time bucket. This forward sweep is efficient computationally in the presence of risk factor and or portfolio path dependence, and is very memory efficient, as only the model state vector X_u and fixings prior to time u that are needed for future exposures are kept in memory at one point in time.

We now express CVA as the solution to an integral, and in an attempt to do so cleary, write the payoff as a function of the entire risk factor set, which are written as functions of the entire set of normally distributed random variables, which in turn are written as function of the entire set of uniform distributed random variables. We do not propose to carry the calculation out in this way due to large memory requirements. Define $\mathbf{X}$ as the matrix of risk factor values at all future required time steps:

$$\mathbf{X} = \left[X_{t_0^f}, X_{t_1^f}, \ldots, X_{t_{n^{\text{fixing}}}^f}\right] \tag{12}$$

and $\boldsymbol{\xi}$ as the stacked $d = n^{\mathtt{factor}} n^{\mathtt{fixing}}$ vector of standard normal random variables:

$$\boldsymbol{\xi} = \left[\xi_1^T, \xi_2^T, \ldots, \xi_{n^{\text{fixing}}}^T\right]^T \tag{13}$$

Define the function from the normal random variables $\boldsymbol{\xi}$ to the risk factor values $\mathbf{X}$ as:

$$\mathbf{X} = g(\boldsymbol{\xi}) \tag{14}$$

where we suppress the dependence on the model parameters A, f and Σ, and initial risk factor values X_0 for notational convenience. Define the CVA payoff in equation 6 as a function π of the risk factor matrix $\mathbf{X}$:

$$\pi(\mathbf{X}) \equiv \sum_{i=1}^{N-1} (1 - R_c) \max(V_{t_i^e} - C_{t_i^e}, 0) D_{t,t_i^e} (S^c_{t,t_{i-1}^e} - S^c_{t,t_i^e}) S^b_{t,t_i^e} \tag{15}$$

Now, by expressing the standard normal in terms of the $d = n^{\mathtt{factor}} n^{\mathtt{fixing}}$ vector of uniform random variables $\mathbf{u} = \left[\mathrm{u}_1^T, \mathrm{u}_2^T, \ldots, \mathrm{u}_{n^{\text{fixing}}}^T\right]^T$ we can explicitly express CVA as an integral of the payoff multiplied by the multi-variable uniform distribution:

$$\begin{aligned} \hat{\mathrm{CVA}}_t &= \int_{\mathbb{R}^d} \pi(\mathbf{X}) \mathrm{PDF}(\mathbf{X}) d\boldsymbol{X} \\ &= \int_{\mathbb{R}^d} \pi(g(\boldsymbol{\xi})) \frac{\exp(-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi})}{\sqrt{2\pi}} d\boldsymbol{\xi} \\ &= \int_{[0,1)^d} \pi(g(\Phi^{-1}(\mathbf{u}))) d\mathbf{u} \end{aligned} \tag{16}$$

where $\mathrm{PDF}(\mathbf{X})$ is the probability distribution function of $\mathbf{X}$, and $\Phi^{-1}(\mathbf{u})$ is the inverse cumulative normal function that operates element wise on each uniform random variable in $\mathbf{u}$.

A.1 Classical Monte Carlo Method (MC)

At the heart of the Monte Carlo method is the approximation of the above expectation (integral) using:

$$\hat{\mathrm{CVA}} \approx \hat{\mathrm{CVA}}_{\mathtt{MC}} = \frac{1}{n} \sum_{\omega=0}^{n-1} \pi(g(\Phi^{-1}(\mathbf{u}^\omega))) \tag{17}$$

where $\mathbf{u}^\omega$ is the ω^{th} realization of $d = n^{\mathtt{factor}} n^{\mathtt{fixing}}$ vector of uniformly distributed pseudo random variables, independent from each other and all earlier draws[2]. 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.

[2] A commonly used pseudo random number generator is the Mersenne Twister algorithm [14], and is what we use throughout this paper

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:

$$\sigma_{\texttt{MC}} = \frac{\sigma_{\texttt{Payoff}}}{\sqrt{n}} \tag{18}$$

Furthermore, the central limit theorem guarantees that the distribution of the MC estimate converges to the normal distribution as the number of replicas approaches ∞:

$$\sqrt{n}\left(\hat{\text{CVA}}_{\texttt{MC}} - \hat{\text{CVA}}\right) \xrightarrow{d} \mathcal{N}\left(0,\, \sigma^2_{\texttt{Payoff}}\right) \tag{19}$$

where $\mathcal{N}(\mu, \sigma^2)$ is the normal distribution with mean μ and variance σ^2. Stated another way:

$$\left(\hat{\text{CVA}}_{\texttt{MC}} - \hat{\text{CVA}}_t\right) \approx \mathcal{N}\left(0,\, \frac{\sigma^2_{\texttt{Payoff}}}{n}\right) \tag{20}$$

and the approximation gets better as we use more replicas n. This allows us to place confidence intervals around the MC estimates.

In practice, σ^2_{Payoff} is not known, but can be estimated using the sample variance:

$$\sigma^2_{\texttt{Payoff}} \approx \frac{1}{n-1}\sum_{\omega=0}^{n-1}\left(\pi(g(\Phi^{-1}(\mathbf{u}^{\omega}))) - \hat{\text{CVA}}_{\texttt{MC}}\right)^2 \tag{21}$$

A.2 Antithetic Sampling

Antithetic sampling attempts to reduce the variance of the Monte Carlo estimator by introducing negative dependence between pairs of random replications.

Assume n is an even number. Draw $n/2$ pseudo random uniform variables $\mathbf{u}^0$, $\mathbf{u}^1$, $\mathbf{u}^2$, …, $\mathbf{u}^{n/2-1}$. Take the mirror image of these samples, $1-\mathbf{u}^0$, 1-$\mathbf{u}^1$, 1-$\mathbf{u}^2$, …, 1-$\mathbf{u}^{n/2-1}$, also valid random uniform random variables, and append them to the end of the first regular pseudo uniform variables to form a sequence with n replications:

$$\begin{aligned}\mathbf{u}^0_{\texttt{AMC}}, \mathbf{u}^1_{\texttt{AMC}}, \mathbf{u}^2_{\texttt{AMC}}, \ldots, \mathbf{u}^{n-1}_{\texttt{AMC}} = \mathbf{u}^0, \mathbf{u}^1, \mathbf{u}^2, \ldots, \mathbf{u}^{n/2-1},\\ 1-\mathbf{u}^0, 1-\mathbf{u}^1, 1-\mathbf{u}^2, \ldots, 1-\mathbf{u}^{n/2-1}\end{aligned} \tag{22}$$

The payoff is then evaluated at each of these points and averaged:

$$\hat{\text{CVA}}_{\texttt{AMC}} = \frac{1}{n}\sum_{\omega=0}^{n-1}\pi(g(\Phi^{-1}(\mathbf{u}^{\omega}_{\texttt{AMC}}))) \tag{23}$$

The standard error formula (18) relies on the n samples being independent, something we intentionally broke by making antithetic pairs $\mathbf{u}^\omega$ and $1-\mathbf{u}^\omega$ perfectly negatively correlated. This can be resolved by averaging the payoff of the antithetic pairs first. The resulting $n/2$ samples are independent, allowing us to use the regular formula in (18) with a reduced set of replications:

$$\sigma_{\texttt{AMC}} = \frac{\sigma_{\texttt{Pairs}}}{\sqrt{\frac{n}{2}}} \tag{24}$$

where

$$\sigma^2_{\texttt{Pairs}} \quad = \quad \frac{1}{n/2-1} \sum_{\omega=0}^{\omega=n/2-1} \left(\frac{1}{2} \left(\pi'(\mathbf{u}^\omega) + \pi'(1-\mathbf{u}^\omega) \right) - \hat{\text{CVA}}_{\texttt{AMC}} \right)^2 \tag{25}$$

where $\pi'(\mathbf{u})$ is shorthand for the CVA payoff as a function of the uniforms. Comparing equation (18) and (24), we can see that antithetic variates asymptotic convergence rate is the same as the MC method $\mathcal{O}(n^{-\frac{1}{2}})$, and that the variance is reduced if and only if:

$$\sigma_{\texttt{Pairs}}\sqrt{2} < \sigma_{\texttt{Payoff}} \tag{26}$$

This occurs if and only if:

$$\text{Cov}\left[\pi'\left(\mathbf{u}\right), \pi'\left(1-\mathbf{u}\right)\right] < 0 \tag{27}$$

A.3 Quasi-Monte Carlo

A low discrepancy sequence is used to generate n replications of a d dimension vector of uniform independent random variables

$$\mathbf{u}^0_{\texttt{LDS}}, \mathbf{u}^1_{\texttt{LDS}}, \ldots, \mathbf{u}^{n-1}_{\texttt{LDS}} \tag{28}$$

The remaining mechanics of the QMC method are identical to those of PMC:

$$\hat{\text{CVA}}_{\texttt{QMC}} = \frac{1}{n} \sum_{\omega=0}^{n-1} \pi(g(\Phi^{-1}(\mathbf{u}^\omega_{\texttt{LDS}}))) \tag{29}$$

The Koksma-Hlawka inequality provides an upper bound on the integration error

$$\epsilon_{\texttt{QMC}} \leq \mathcal{O}\left(\frac{\ln^d(n)}{n}\right), \tag{30}$$

but it turns out that for most practical applications and reasonable sizes of n and $d > 1$, the asymptotic error bound grossly overestimates the actual size of the error. Instead, what has been repeatedly observed is a power law [5], [10], [1]:

$$\epsilon_{\mathtt{QMC}} \approx \alpha n^{-\beta}. \tag{31}$$

where β is close to $1/2$ for payoffs with high effective dimension and is close to 1 for payoffs with effective dimension equal to 1. An effective dimension d^e roughly states that a function can be approximated by a sum of functions with dimension equal to d^e and below. See Caflisch et. al. 1997 [5] for more details. It turns out, that for many payoffs encountered in practice, the effective dimension is significantly less than the nominal dimension.

Of the low discrepancy sequences available, Sobol sequences [16] have been shown to have superior convergence properties, to be the most robust to the number of dimensions, and 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 [13]). We use direction numbers provided by Broda commercially that allow for the generation of 65,536 dimensions [3]. The direction numbers, constructed in conjunction with Ilya Sobol, were chosen such that all 65K dimensions satisfy property A, and that all adjacent five dimensions satisfy property 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 firsts 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 split them into four equal sub-intervals.

A.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:

$$\hat{W}_t = \hat{W}_s + \frac{t-s}{u-s}\left(\hat{W}_u - \hat{W}_s\right) + \xi\sqrt{\frac{(t-s)(u-t)}{u-s}} \tag{32}$$

to simulate each independent Brownian motion in a non-chronological order such that each step explains a maximum amount of remaining variation. See Caflisch et al. 1997 [5] for the original description. The Brownian motion paths depend more on earlier draws of the random number generator than later ones. Each successive random draw adds finer and finer details to the path. This has two effects. First, it reduces the effective dimension, in that a certain percentage of the variation of the path can be explained by fewer uniform random variables. Sobol and Kucherenko 2005 [17] apply global sensitivity analysis to illustrate the percentage variation of the Brownian path captured by each uniform random variable. The plot, contained in Figure 1 on page 6, clearly shows that variation explained by each uniform drops off quickly when using the Brownian bridge discretization, and very slowly when using the classical discretization.

The second effect, related to the first, is that the path depends more on earlier random variables from the sequences than later ones, which for some LDS generators matters as they are generated such that discrepancy of the two-dimensional projections is lower for random variables created earlier in the sequence.

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 $i = 1, 2, \ldots, n^f$:

$$W_{t_i^f} = W_{t_{i-1}^f} + \sqrt{R}\left[\hat{W}_{t_i^f} - \hat{W}_{t_{i-1}^f}\right] \tag{33}$$

$$X_{t_i^f} = X_{t_{i-1}^f} + \left[AX_{t_{i-1}^f} + f_{t_{i-1}^f}\right]\Delta_i + \Sigma_{t_{i-1}^f}\left[W_{t_i^f} - W_{t_{i-1}^f}\right] \tag{34}$$

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:

$$\mathbf{X}_{\texttt{BB}} = \text{BB}(\boldsymbol{\xi}) \tag{35}$$

The Brownian bridge QMC CVA estimate is then equal to:

$$\hat{\text{CVA}}_{\texttt{QMC,BB}} = \frac{1}{n}\sum_{\omega=0}^{n-1} \pi(\text{BB}(\Phi^{-1}(\mathbf{u}_{\texttt{LDS}}^{\omega}))) \tag{36}$$

A.5 Randomized Quasi-Monte Carlo

In the classical Monte Carlo framework, the standard error over n paths is given by 18 or 24, 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 quasi-random sequence, no formula of practical use exists for calculating the standard error.

Randomized Quasi-Monte Carlo (RQMC) is a technique for obtaining independent estimates from multiple QMC runs for the purpose of estimating the standard error of the mean simulation result. To accomplish this, a quasi-random sequence $\{\mathbf{u}_{\texttt{LDS}}^i\}_{i=1}^{n/m}$ is randomized k times to produce k independent sequences $\{\mathbf{y}_i^1\}, \ldots, \{\mathbf{y}_i^m\}$, where $i = 1, \ldots, n/k$, where we assume n divides evenly into k. Performing QMC simulations of the function f over k independent trials using these sequences leads to estimates $\hat{I}^1_{n/k}(f), \ldots, \hat{I}^k_{n/k}(f)$ that can be averaged to obtain the final result:

$$\bar{I}^k_n(f) = \frac{1}{k}\sum_{j=1}^{k} \hat{I}^j_{n/k}(f). \tag{37}$$

Because $\{\hat{I}^j_{n/k}\}$ are mutually independent, the standard error of the mean can be calculated via

$$\bar{\epsilon}_{k,n} = \frac{\bar{\sigma}_{k,n}}{\sqrt{k}}, \tag{38}$$

where

$$\bar{\sigma}^2_{k,n} = \frac{1}{k-1}\sum_{j=1}^{k} \left(\hat{I}^j_n(f) - \bar{I}^k_n(f)\right)^2. \tag{39}$$

Several techniques exist for randomizing $\{\mathbf{u}^i_{\mathtt{LDS}}\}_{i=1}^{n/k}$ to obtain $\{\mathbf{y}^j_i\}_{i=1}^{n/k}$ for $j = 1, 2, \ldots, k$. In this paper, we use *digital shift*. Belonging to the class of *scrambling* techniques, this method randomly switches the bits in the binary representation of the elements in the sequence and preserves the discrepancy of the sequence:

$$\mathbf{y}^j_i = \mathbf{u}^i_{\mathtt{LDS}} \oplus \mathbf{u}^j \tag{40}$$

where $\mathbf{u}^j$ are independently drawn from the d-dimensional uniform distribution in the $[0, 1)$ interval. See Glasserman 2004 [10] or Giles et al. [9] for more information.

B A proxy for equivalent paths

For each method, CVA is estimated using $n_0 < n_1 < \ldots < n_{l-1}$ simulations (all powers of two to ensure smooth convergence of QMC). We repeat this set of calculations across m trials, where each trial uses a non-overlapping sequence of the random numbers, similar to the test setup in Morokoff and Caflisch 1995 [15], and Bianchetti, Kucherenko, and Scoleri 2015 [1]. Denote $I^k_n(f, \mathcal{M})$ as the estimate of the expected value of the payoff f using n paths, trial number k and methodology $\mathcal{M}$. We define the benchmark value for a particular payoff and methodology as the average of the m estimates using the largest number of simulations n_{l-1} :

$$I_m(f, \mathcal{M}) \quad = \quad \frac{1}{m}\sum_{k=0}^{m-1} I^k_{n_{l-1}}(f, \mathcal{M}) \tag{41}$$

The error for each trial is then computed by subtracting the benchmark value from the approximation

$$e_n^k(f,\mathcal{M}) = I_n^k(f,\mathcal{M}) - I_m(f,\mathcal{M}), \tag{42}$$

providing an empirical distribution of the errors for each payoff, number of simulations, and methodology. The root mean squared error (RMSE) for a particular number of scenarios n, methodology $\mathcal{M}$, and payoff f is defined as the sample standard deviation:

$$\mathrm{RMSE}(n,f,\mathcal{M}) = \sqrt{\frac{1}{m-1}\sum_{k=0}^{m-1} e_n^k(f,\mathcal{M})^2} \tag{43}$$

Assume the RMSE follows a power law of convergence:

$$\mathrm{RMSE}(n,f,\mathcal{M}) = \alpha(f,\mathcal{M})n^{-\beta(f,\mathcal{M})}. \tag{44}$$

This is a fact for classical MC, see Glasserman 2004 [10] for example, and has been repeatedly observed in numerical experiments for QMC, see Morokoff and Caflisch 1995 [15], Caflisch, Morokoff, and Owen 1997 [5], and Bianchetti, Kucherenko, and Scoleri 2015 [1] for example.

It is known that classical MC has an error that is normally distributed with mean 0 and standard deviation equal to the standard deviation of the payoff divided by the square root of the number of simulation paths:

$$\mathrm{RMSE}(n,f,\mathtt{MC}) = \sigma_f n^{-\frac{1}{2}} \tag{45}$$

where f is the payoff and σ_f the standard deviation of the payoff. The coefficient σ_f can be estimated in sample as the sample standard deviation of the payoff. We can also see from the definition of the RMSE, that this is also the case even when using QMC. To see, take n to be 1, then the RMSE is equal to the square root of the expected value of the payoff less the payoff mean, squared, which is the QMC estimate of the variance of the payoff.

$$\mathrm{RMSE}(n,f,\mathtt{QMC}) = \sigma_f n^{-\beta(f,\mathtt{QMC})} \tag{46}$$

The convergence rate β can then be estimated using linear least squares regression of the log RMSE onto the log number of scenarios n with a known intercept:

$$\hat{\beta} = \frac{\sum_{i=0}^{l-1}\ln(n_i)\left(\ln(\mathrm{RMSE}(n_i)) - \ln(\sigma_f)\right)}{\sum_{i=0}^{l-1}\ln(n_i)^2}, \tag{47}$$

where we have suppressed the payoff f and method $\mathcal{M}$ arguments for notational convenience.

For randomized quasi MC (RQMC), we assume the RMSE for each of the k trials as a function of the number of simulations per trial follow the same power law with the same coefficients, leading to:

$$\mathrm{RMSE}(n, f, \texttt{RQMC}) = \frac{\sigma_f}{k^{\frac{1}{2}}} \frac{n}{k}^{-\beta(f,\texttt{RQMC})} \quad (48)$$
$$= k^{\beta-\frac{1}{2}} \sigma_f n^{-\beta(f,\texttt{RQMC})} \quad (49)$$

Beta can then be estimated by regressing the log RMSE onto the adjusted number of paths $\ln(n) - \ln(k)$ with the known intercept of $\ln(\sigma_f) - .5\ln(k)$. Note that if we assume that the convergence rate is the same as regular QMC (something we have repeatedly observed for practical cases, but in some cases it has been shown that RQMC can actually improve the convergence rate [10].), the RQMC RMSE is $k^{\beta-\frac{1}{2}}$ times larger than QMC RMSE.

Equivalent paths between two methods, say QMC and MC, is defined as the number of QMC paths p needed to match the RMSE with n MC paths:

$$\sigma_f n^{-\frac{1}{2}} = \alpha(f, \texttt{QMC}) p^{-\beta(f,\texttt{QMC})} \quad (50)$$

which can be solved for as

$$\mathrm{p}(n, f, \texttt{QMC}, \texttt{MC}) = \left(\frac{\mathrm{RMSE}(n, f, \texttt{MC})}{\alpha(f, \texttt{QMC})} \right)^{-\frac{1}{\beta(f,\texttt{QMC})}} \quad (51)$$

In our tests, we use the following procedure for each portfolio and each calculation type $f \in$ {CVA, CR Delta, IR Delta, IR Vega, FX Delta, and FX Vega}: First estimate σ_f using 51,200 paths, second use $n_0 = 32, n_1 = 1,024, n_2 = 4,096, n_3 = 16,384$ QMC simulations, $m = 50$ trials, and equation 43 to estimate $\mathrm{RMSE}(32, f, \texttt{QMC})$, $\mathrm{RMSE}(1024, f, \texttt{QMC})$, $\mathrm{RMSE}(4096, f, \texttt{QMC})$, and $\mathrm{RMSE}(16384, f, \texttt{QMC})$, third, use the estimated RMSEs and the least squares equation 47 to estimate the convergence coefficient $\beta(f, \texttt{QMC})$, and fourth, use the computed coefficients and equation 51 and 18 to compute the number of QMC paths required to produce errors equivalent to the errors produced by 10,000 classical MC paths.

C Supporting Results

A summary of the RMSEs for MC, MC with antithetic sampling, QMC, QMC + BB, and RQMC using 1024 paths are presented in tables 8, 9 10, 11, and 12 respectively. Note that when there would be more than one result for a particular entry in the matrix, such as IR and FX Deltas and Vegas for the six and eleven currency portfolios, we aggregate by simply adding the results. This is equivalent to the total RMSE of the sum of the constituent elements when corresponding errors are perfectly correlated.

A summary of the estimated QMC, QMC + BB, RQMC + BB convergence rates β are presented in tables 13, 14, and 15 respectively. Results are aggregated by averaging the results. As a result, the β provided are not equivalent to the convergence rate of the sum of all of the constituent CVA and CVA sensitivities. This only holds in the case when there is one measurement (CVA, CR Delta, 1 currency IR and FX Delta, 1 currency IR and FX Vega).

A summary of the estimated convergence constants σ_f for MC, QMC, and QMC + BB are presented in table 16. A summary of the convergence constant for AMC and RQMC + BB are presented in table 17 and 18 respectively. Results are aggregated by adding the constituent results.

Table 19 looks at the acceleration from a different angle, holding the number of QMC + BB paths fixed at 1,024, and finding how many classical MC paths are needed to match the error. On average, it takes approximately 68,966 MC paths. The large difference in acceleration (13 versus 67) is a result of the equivalent MC paths increasing faster than the equivalent QMC paths shrink, and thus the average of the former is higher than the average of the latter.

Table 7: CVA and CVA sensitivities using QMC + BB paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread).

Spread	Type	1 CCY	6 CCY	11 CCY	Average
-300bps	CVA	106.55	111.00	111.47	109.68
-300bps	CR Delta	1.47	1.52	1.53	1.51
-300bps	IR, FX Delta	0.27	0.36	0.37	0.33
-300bps	IR, FX Vega	0.05	0.00	0.00	0.02
-100bps	CVA	49.34	44.40	46.87	46.87
-100bps	CR Delta	0.62	0.57	0.60	0.60
-100bps	IR, FX Delta	0.31	0.37	0.37	0.35
-100bps	IR, FX Vega	0.17	0.04	0.07	0.10
0bps	CVA	27.95	17.95	21.73	22.54
0bps	CR Delta	0.31	0.19	0.23	0.24
0bps	IR, FX Delta	0.28	0.33	0.33	0.31
0bps	IR, FX Vega	0.22	0.10	0.13	0.15
100bps	CVA	13.99	4.84	7.65	8.83
100bps	CR Delta	0.13	0.04	0.07	0.08
100bps	IR, FX Delta	0.21	0.17	0.20	0.19
100bps	IR, FX Vega	0.20	0.08	0.11	0.13
300bps	CVA	2.53	0.13	0.47	1.04
300bps	CR Delta	0.02	0.00	0.00	0.01
300bps	IR, FX Delta	0.08	0.01	0.03	0.04
300bps	IR, FX Vega	0.08	0.01	0.02	0.04

Table 8: CVA and CVA sensitivities RMSE estimates using classical MC with 1,024 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates.

Spread	Type	1 CCY	6 CCY	11 CCY	Average
-300bps	CVA	1.572	1.021	1.291	1.295
-300bps	CR Delta	0.018	0.010	0.013	0.014
-300bps	IR, FX Delta	0.008	0.013	0.014	0.012
-300bps	IR, FX Vega	0.014	0.019	0.025	0.019
-100bps	CVA	1.261	0.818	1.029	1.036
-100bps	CR Delta	0.014	0.008	0.011	0.011
-100bps	IR, FX Delta	0.007	0.012	0.013	0.011
-100bps	IR, FX Vega	0.010	0.013	0.018	0.014
0bps	CVA	0.992	0.569	0.755	0.772
0bps	CR Delta	0.010	0.005	0.008	0.008
0bps	IR, FX Delta	0.008	0.011	0.012	0.010
0bps	IR, FX Vega	0.009	0.009	0.013	0.010
100bps	CVA	0.691	0.258	0.435	0.461
100bps	CR Delta	0.006	0.003	0.004	0.004
100bps	IR, FX Delta	0.008	0.009	0.010	0.009
100bps	IR, FX Vega	0.008	0.006	0.010	0.008
300bps	CVA	0.246	0.017	0.094	0.119
300bps	CR Delta	0.002	0.000	0.001	0.001
300bps	IR, FX Delta	0.005	0.002	0.004	0.004
300bps	IR, FX Vega	0.006	0.002	0.004	0.004

Table 9: CVA and CVA sensitivities RMSE estimates using antithetic MC with 1,024 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates.

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	0.288	0.254	0.360	0.301
-300bps	CR Delta	0.003	0.002	0.003	0.003
-300bps	IR, FX Delta	0.003	0.006	0.006	0.005
-300bps	IR, FX Vega	0.008	0.010	0.014	0.010
-100bps	CVA	0.585	0.295	0.421	0.434
-100bps	CR Delta	0.006	0.003	0.004	0.004
-100bps	IR, FX Delta	0.003	0.006	0.006	0.005
-100bps	IR, FX Vega	0.009	0.010	0.013	0.010
0bps	CVA	0.689	0.395	0.508	0.530
0bps	CR Delta	0.007	0.004	0.005	0.005
0bps	IR, FX Delta	0.003	0.007	0.007	0.006
0bps	IR, FX Vega	0.008	0.008	0.011	0.009
100bps	CVA	0.614	0.280	0.402	0.432
100bps	CR Delta	0.006	0.002	0.004	0.004
100bps	IR, FX Delta	0.005	0.008	0.008	0.007
100bps	IR, FX Vega	0.008	0.006	0.009	0.008
300bps	CVA	0.269	0.034	0.087	0.130
300bps	CR Delta	0.002	0.000	0.001	0.001
300bps	IR, FX Delta	0.006	0.002	0.004	0.004
300bps	IR, FX Vega	0.007	0.002	0.004	0.004

Table 10: CVA and CVA sensitivities RMSE estimates using QMC with 1,024 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates.

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	0.185	0.200	0.307	0.231
-300bps	CR Delta	0.002	0.002	0.003	0.002
-300bps	IR, FX Delta	0.001	0.004	0.005	0.003
-300bps	IR, FX Vega	0.003	0.007	0.011	0.007
-100bps	CVA	0.216	0.199	0.404	0.273
-100bps	CR Delta	0.002	0.002	0.004	0.002
-100bps	IR, FX Delta	0.001	0.005	0.006	0.004
-100bps	IR, FX Vega	0.003	0.007	0.011	0.007
0bps	CVA	0.222	0.194	0.406	0.274
0bps	CR Delta	0.002	0.002	0.004	0.002
0bps	IR, FX Delta	0.002	0.006	0.006	0.005
0bps	IR, FX Vega	0.003	0.006	0.009	0.006
100bps	CVA	0.273	0.161	0.320	0.252
100bps	CR Delta	0.003	0.001	0.003	0.002
100bps	IR, FX Delta	0.002	0.005	0.007	0.005
100bps	IR, FX Vega	0.003	0.004	0.008	0.005
300bps	CVA	0.192	0.030	0.081	0.101
300bps	CR Delta	0.002	0.000	0.001	0.001
300bps	IR, FX Delta	0.003	0.002	0.004	0.003
300bps	IR, FX Vega	0.004	0.001	0.004	0.003

Table 11: CVA and CVA sensitivities RMSE estimates using QMC + BB with 1,024 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates.

Spread	Type	1 CCY	6 CCY	11 CCY	Average
-300bps	CVA	0.110	0.097	0.127	0.111
-300bps	CR Delta	0.001	0.001	0.001	0.001
-300bps	IR, FX Delta	0.001	0.002	0.002	0.002
-300bps	IR, FX Vega	0.001	0.003	0.004	0.003
-100bps	CVA	0.088	0.073	0.117	0.093
-100bps	CR Delta	0.001	0.001	0.001	0.001
-100bps	IR, FX Delta	0.001	0.002	0.003	0.002
-100bps	IR, FX Vega	0.001	0.003	0.004	0.003
0bps	CVA	0.079	0.066	0.120	0.088
0bps	CR Delta	0.001	0.001	0.001	0.001
0bps	IR, FX Delta	0.001	0.003	0.003	0.002
0bps	IR, FX Vega	0.001	0.002	0.004	0.003
100bps	CVA	0.060	0.047	0.081	0.063
100bps	CR Delta	0.001	0.000	0.001	0.001
100bps	IR, FX Delta	0.001	0.003	0.003	0.002
100bps	IR, FX Vega	0.001	0.002	0.003	0.002
300bps	CVA	0.036	0.014	0.043	0.031
300bps	CR Delta	0.000	0.000	0.000	0.000
300bps	IR, FX Delta	0.001	0.001	0.002	0.001
300bps	IR, FX Vega	0.001	0.001	0.002	0.001

Table 12: CVA and CVA sensitivities RMSE estimates using RQMC + BB ($k = 16$ trials) with 1,024 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates.

Spread	Type	1 CCY	6 CCY	11 CCY	Average
-300bps	CVA	0.358	0.286	0.401	0.348
-300bps	CR Delta	0.004	0.003	0.004	0.004
-300bps	IR, FX Delta	0.002	0.004	0.006	0.004
-300bps	IR, FX Vega	0.003	0.007	0.011	0.007
-100bps	CVA	0.303	0.224	0.331	0.286
-100bps	CR Delta	0.003	0.002	0.003	0.003
-100bps	IR, FX Delta	0.002	0.005	0.005	0.004
-100bps	IR, FX Vega	0.003	0.006	0.010	0.006
0bps	CVA	0.275	0.171	0.298	0.248
0bps	CR Delta	0.003	0.002	0.003	0.003
0bps	IR, FX Delta	0.002	0.005	0.006	0.004
0bps	IR, FX Vega	0.003	0.005	0.009	0.006
100bps	CVA	0.234	0.109	0.214	0.185
100bps	CR Delta	0.002	0.001	0.002	0.002
100bps	IR, FX Delta	0.002	0.004	0.006	0.004
100bps	IR, FX Vega	0.003	0.004	0.007	0.004
300bps	CVA	0.121	0.032	0.062	0.072
300bps	CR Delta	0.001	0.000	0.000	0.001
300bps	IR, FX Delta	0.002	0.002	0.003	0.002
300bps	IR, FX Vega	0.003	0.002	0.003	0.002

Table 13: Approximate CVA and CVA sensitivities convergence rates β using QMC for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread).

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	0.80	0.70	0.71	0.74
-300bps	CR Delta	0.82	0.72	0.73	0.76
-300bps	IR, FX Delta	0.75	0.67	0.65	0.69
-300bps	IR, FX Vega	0.68	0.63	0.62	0.64
-300bps	Average	0.76	0.68	0.68	0.71
-100bps	CVA	0.74	0.67	0.69	0.70
-100bps	CR Delta	0.75	0.69	0.71	0.71
-100bps	IR, FX Delta	0.76	0.63	0.62	0.67
-100bps	IR, FX Vega	0.67	0.58	0.58	0.61
-100bps	Average	0.73	0.64	0.65	0.67
0bps	CVA	0.71	0.63	0.66	0.66
0bps	CR Delta	0.73	0.64	0.68	0.68
0bps	IR, FX Delta	0.70	0.59	0.60	0.63
0bps	IR, FX Vega	0.68	0.55	0.55	0.59
0bps	Average	0.70	0.60	0.62	0.64
100bps	CVA	0.63	0.56	0.60	0.60
100bps	CR Delta	0.63	0.56	0.60	0.60
100bps	IR, FX Delta	0.67	0.55	0.57	0.60
100bps	IR, FX Vega	0.67	0.53	0.54	0.58
100bps	Average	0.65	0.55	0.58	0.59
300bps	CVA	0.54	0.53	0.51	0.53
300bps	CR Delta	0.54	0.52	0.51	0.53
300bps	IR, FX Delta	0.56	0.51	0.51	0.53
300bps	IR, FX Vega	0.55	0.51	0.50	0.52
300bps	Average	0.55	0.52	0.51	0.53
	Average	0.68	0.60	0.61	0.63

Table 14: Approximate CVA and CVA sensitivities convergence rates β using QMC + BB for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread).

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	0.90	0.85	0.78	0.85
-300bps	CR Delta	0.90	0.85	0.80	0.85
-300bps	IR, FX Delta	0.87	0.81	0.75	0.81
-300bps	IR, FX Vega	0.83	0.78	0.70	0.77
-300bps	Average	0.87	0.82	0.76	0.82
-100bps	CVA	0.89	0.82	0.81	0.84
-100bps	CR Delta	0.89	0.82	0.82	0.84
-100bps	IR, FX Delta	0.88	0.74	0.71	0.78
-100bps	IR, FX Vega	0.83	0.70	0.65	0.73
-100bps	Average	0.87	0.77	0.75	0.80
0bps	CVA	0.87	0.78	0.73	0.79
0bps	CR Delta	0.87	0.76	0.74	0.79
0bps	IR, FX Delta	0.84	0.70	0.68	0.74
0bps	IR, FX Vega	0.85	0.66	0.62	0.71
0bps	Average	0.86	0.72	0.70	0.76
100bps	CVA	0.84	0.75	0.69	0.76
100bps	CR Delta	0.83	0.73	0.69	0.75
100bps	IR, FX Delta	0.81	0.65	0.66	0.70
100bps	IR, FX Vega	0.84	0.63	0.61	0.69
100bps	Average	0.83	0.69	0.66	0.73
300bps	CVA	0.77	0.63	0.61	0.67
300bps	CR Delta	0.76	0.62	0.61	0.66
300bps	IR, FX Delta	0.75	0.60	0.59	0.64
300bps	IR, FX Vega	0.74	0.57	0.56	0.63
300bps	Average	0.76	0.61	0.59	0.65
	Average	0.84	0.72	0.69	0.75

Table 15: Approximate CVA and CVA sensitivities convergence rates β using RQMC + BB for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread).

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	0.89	0.82	0.82	0.84
-300bps	CR Delta	0.88	0.82	0.83	0.84
-300bps	IR, FX Delta	0.87	0.80	0.77	0.82
-300bps	IR, FX Vega	0.85	0.77	0.74	0.78
-300bps	Average	0.87	0.80	0.79	0.82
-100bps	CVA	0.88	0.81	0.80	0.83
-100bps	CR Delta	0.88	0.81	0.81	0.83
-100bps	IR, FX Delta	0.88	0.76	0.73	0.79
-100bps	IR, FX Vega	0.82	0.71	0.68	0.74
-100bps	Average	0.86	0.74	0.76	0.80
0bps	CVA	0.85	0.79	0.77	0.80
0bps	CR Delta	0.85	0.79	0.77	0.80
0bps	IR, FX Delta	0.86	0.73	0.69	0.76
0bps	IR, FX Vega	0.83	0.66	0.64	0.71
0bps	Average	0.85	0.74	0.72	0.77
100bps	CVA	0.81	0.74	0.73	0.76
100bps	CR Delta	0.80	0.73	0.72	0.75
100bps	IR, FX Delta	0.83	0.67	0.66	0.72
100bps	IR, FX Vega	0.82	0.63	0.61	0.69
100bps	Average	0.81	0.69	0.68	0.73
300bps	CVA	0.73	0.61	0.60	0.65
300bps	CR Delta	0.72	0.61	0.59	0.64
300bps	IR, FX Delta	0.72	0.60	0.59	0.64
300bps	IR, FX Vega	0.72	0.56	0.56	0.61
300bps	Average	0.72	0.59	0.58	0.63
	Average	0.82	0.72	0.71	0.75

Table 16: Approximate CVA and CVA sensitivities convergence constant α (equal to the payoff standard deviation σ_f) using classical MC for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread). Results are aggregated by adding the constituent alphas in the group.

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	54.58	30.34	40.47	41.80
-300bps	CR Delta	0.58	0.32	0.43	0.44
-300bps	IR, FX Delta	0.25	0.43	0.45	0.37
-300bps	IR, FX Vega	0.45	0.60	0.79	0.61
-100bps	CVA	41.62	23.74	31.50	32.29
-100bps	CR Delta	0.45	0.25	0.34	0.35
-100bps	IR, FX Delta	0.24	0.40	0.40	0.35
-100bps	IR, FX Vega	0.31	0.43	0.57	0.44
0bps	CVA	30.87	17.10	23.97	23.98
0bps	CR Delta	0.32	0.18	0.25	0.25
0bps	IR, FX Delta	0.25	0.36	0.37	0.33
0bps	IR, FX Vega	0.28	0.30	0.42	0.33
100bps	CVA	20.74	9.29	15.02	15.01
100bps	CR Delta	0.20	0.08	0.14	0.14
100bps	IR, FX Delta	0.24	0.28	0.32	0.28
100bps	IR, FX Vega	0.27	0.20	0.31	0.26
300bps	CVA	8.27	1.25	2.76	4.10
300bps	CR Delta	0.07	0.01	0.02	0.03
300bps	IR, FX Delta	0.16	0.08	0.12	0.12
300bps	IR, FX Vega	0.20	0.06	0.11	0.12

Table 17: Approximate CVA and CVA sensitivities convergence constant α (equal to the payoff pair standard deviation σ_f^p) using antithetic MC for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread). Results are aggregated by adding the constituent alphas in the group.

Spread	**Type**	**1 CCY**	**6 CCY**	**11 CCY**	**Average**
-300bps	CVA	9.22	8.11	11.53	9.62
-300bps	CR Delta	0.09	0.08	0.11	0.09
-300bps	IR, FX Delta	0.09	0.18	0.20	0.16
-300bps	IR, FX Vega	0.24	0.32	0.44	0.33
-100bps	CVA	18.73	9.43	13.49	13.88
-100bps	CR Delta	0.19	0.09	0.13	0.13
-100bps	IR, FX Delta	0.09	0.19	0.20	0.16
-100bps	IR, FX Vega	0.28	0.31	0.40	0.33
0bps	CVA	22.04	12.63	16.25	16.97
0bps	CR Delta	0.23	0.13	0.17	0.17
0bps	IR, FX Delta	0.11	0.24	0.23	0.19
0bps	IR, FX Vega	0.24	0.27	0.36	0.29
100bps	CVA	19.64	8.97	12.87	13.83
100bps	CR Delta	0.19	0.08	0.12	0.13
100bps	IR, FX Delta	0.17	0.26	0.26	0.23
100bps	IR, FX Vega	0.24	0.20	0.28	0.24
300bps	CVA	8.60	1.08	2.79	4.16
300bps	CR Delta	0.07	0.01	0.02	0.03
300bps	IR, FX Delta	0.18	0.07	0.12	0.12
300bps	IR, FX Vega	0.22	0.06	0.12	0.13

Table 18: Approximate CVA and CVA sensitivities convergence constant $\alpha(f, \texttt{RQMCBB})$ using RQMC + BB for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread). Results are aggregated by adding the constituent alphas in the group.

Spread	Type	1 CCY	6 CCY	11 CCY	Average
-300bps	CVA	171.63	89.00	111.24	123.96
-300bps	CR Delta	1.83	0.96	1.19	1.33
-300bps	IR, FX Delta	0.76	1.07	0.94	0.92
-300bps	IR, FX Vega	1.26	1.43	1.54	1.41
-100bps	CVA	130.55	68.87	81.05	93.49
-100bps	CR Delta	1.41	0.75	0.89	1.02
-100bps	IR, FX Delta	0.73	0.88	0.76	0.79
-100bps	IR, FX Vega	0.91	0.86	0.93	0.90
0bps	CVA	97.35	46.05	53.66	65.69
0bps	CR Delta	1.00	0.46	0.57	0.68
0bps	IR, FX Delta	0.73	0.73	0.64	0.70
0bps	IR, FX Vega	0.86	0.53	0.63	0.67
100bps	CVA	61.96	20.05	27.83	36.61
100bps	CR Delta	0.58	0.17	0.25	0.33
100bps	IR, FX Delta	0.70	0.51	0.52	0.58
100bps	IR, FX Vega	0.81	0.33	0.46	0.53
300bps	CVA	16.92	0.91	2.13	6.65
300bps	CR Delta	0.13	0.01	0.02	0.05
300bps	IR, FX Delta	0.38	0.07	0.12	0.19
300bps	IR, FX Vega	0.45	0.05	0.10	0.20

Table 19: Approximate number of classical MC paths (in thousands) needed to produce CVA and CVA sensitivities with errors roughly equivalent to QMC + BB with 1,024 paths for portfolios of various sizes (one 10 year fixed rate payer swap in each currency) and fixed swap rates (set to par plus the given spread). Details of the equivalent path methodology can be found in appendix B.

Spread	Type	1 CCY	6 CCY	11 CCY	Average
-300bps	CVA	245	98	101	148
-300bps	CR Delta	224	93	93	137
-300bps	IR, FX Delta	176	80	56	104
-300bps	IR, FX Vega	108	54	34	66
-300bps	Average	188	81	71	114
-100bps	CVA	223	105	73	134
-100bps	CR Delta	206	89	66	120
-100bps	IR, FX Delta	218	42	33	97
-100bps	IR, FX Vega	103	26	16	48
-100bps	Average	187	65	47	100
0bps	CVA	151	68	40	86
0bps	CR Delta	128	54	42	75
0bps	IR, FX Delta	128	23	22	58
0bps	IR, FX Vega	114	14	12	47
0bps	Average	130	40	29	67
100bps	CVA	120	39	35	65
100bps	CR Delta	98	30	33	54
100bps	IR, FX Delta	70	13	14	32
100bps	IR, FX Vega	102	9	9	40
100bps	Average	97	23	23	48
300bps	CVA	53	8	4	22
300bps	CR Delta	40	8	4	17
300bps	IR, FX Delta	36	5	4	15
300bps	IR, FX Vega	40	4	3	16
300bps	Average	42	6	4	17
	Average	129	43	35	69

Table 20: CVA and CVA sensitivities RMSE estimates using QMC + BB with 1,024 paths for a single USD swap with various fixed swap rates (set to par plus the given spread) and global models (modelling in total one, six, eleven, and thirty currencies).

Spread	Type	1 CCY	6 CCY	11 CCY	30 CCY	MC
-300bps	CVA	0.110	0.148	0.145	0.118	1.706
-300bps	CR Delta	0.001	0.002	0.002	0.001	0.018
-300bps	IR, FX Delta	0.001	0.001	0.001	0.001	0.008
-300bps	IR, FX Vega	0.001	0.002	0.003	0.002	0.014
-100bps	CVA	0.088	0.162	0.167	0.159	1.301
-100bps	CR Delta	0.001	0.002	0.002	0.002	0.014
-100bps	IR, FX Delta	0.001	0.001	0.001	0.001	0.007
-100bps	IR, FX Vega	0.001	0.003	0.003	0.002	0.010
0bps	CVA	0.079	0.201	0.213	0.149	0.965
0bps	CR Delta	0.001	0.002	0.003	0.002	0.010
0bps	IR, FX Delta	0.001	0.001	0.001	0.001	0.008
0bps	IR, FX Vega	0.001	0.002	0.002	0.002	0.009
100bps	CVA	0.060	0.147	0.214	0.149	0.648
100bps	CR Delta	0.001	0.001	0.002	0.002	0.006
100bps	IR, FX Delta	0.001	0.003	0.002	0.002	0.008
100bps	IR, FX Vega	0.001	0.002	0.003	0.002	0.008
300bps	CVA	0.036	0.100	0.122	0.112	0.259
300bps	CR Delta	0.000	0.001	0.001	0.001	0.002
300bps	IR, FX Delta	0.001	0.002	0.002	0.002	0.005
300bps	IR, FX Vega	0.001	0.002	0.003	0.003	0.006