Efficient Monte Carlo Methods for CVA

Michael Meyer

18450547

Assignment presented in the partial fulfilment
of the requirement for the course
Mcomm (Financial Risk Management)
December 2019

Lecturer : Dr. CJ van der Merwe

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

The credit valuation adjustment (CVA) is an important credit risk measure which is defined as the present value of the prospective loss of a party due to a counterparty failing to adhere to their contractual obligations. CVA became necessary in 2000 for banks to incorporate across the globe, however Hull and White (2012) states that marking derivative portfolios to market without including counterparty risk was a common practice in the industry prior to the financial crisis of 2007/2008. The financial crises highlited the importance of the CVA charge when profit and loss (PL) fluctuations owing to CVA changes were quantified in billions of dollars. CVA is one metric that is part of the over arching term xVA, which is defined as adjustments to derivatives contract that incorporate measures such as account funding, credit risk and regulatory capital costs to determine the fair value of the contract. The price of a new trade will usually incorporate the costs associated with xVAs. These adjustments are also frequently used for risk management practices. A comprehensive overview of xVA methodologies can be found in Gregory (2020).

1.2 THE CVA CHALLENGE

The computational complexity of CVA modeling is a significant hurdle. The CVA for a portfolio consisting of OTC derivatives is calculated using the portfolio's lifetime exposure. This necessitates determining the price of a substantial amount of derivative contracts at various future time points which could be long dated contracts. It's worth noting that CVA calculations are performed at the netting set level, with the netting set containing a variety of distinct type of contracts and underlying risk factors. When characteristics like netting, margins, collateralization, and wrong way risk are taken into account, keeping the computational effort manageable entails extensive mathematical modeling and efficient implementations. As a result, except for in a few simple circumstances, CVA is estimated using Monte Carlo (MC) simulation, since the portfolio value could potentially be dependant of a variety of market risk factors. Finally, netting effects with current transactions in the same netting set must be taken into consideration when quoting CVA

for a new deal, and the resulting incremental CVA should preferably be priced in real-time.

1.3 MONTE CARLO

Monte Carlo estimation of an expectation in the context of CVA involves randomly sampling the underlying risk factors N times with real world scenarios, while simulating risk neutral default probabilities and averaging the results. The CVA estimate approaches the true expectation with probability one. The error is normally distributed with mean zero and standard deviation equal to the standard deviation of the CVA dived by the square root of the number of simulations N. Glasserman (2004) provides an excellent detailed explanations of MC and its application to finance. The standard error is often required to be on average 100 times smaller than the standard deviation of the CVA payment, which necessitates 10,000 replications, which is a common quantity. This emphasizes MC's primary drawback: its high computational cost. This is especially important in the situation of CVA, because each payout evaluation is also computationally costly.

Consider a bank with 100,000 trades and 100 exposure dates discretized over time. One replication of the CVA payment across all counterparties requires roughly 10,000,000 trade prices (assuming transaction maturities are evenly distributed), thus that one MC CVA estimate using 10,000 paths requires on the order of 100,000,000,000 trade price assessments. Many banks additionally utilize risk management to manage these credit adjustments, which involves calculating the CVA's derivatives in relation to the market prices of the instruments used to hedge it. Bump-and-run strategies necessitate at least one full MC CVA calculation per derivative; 200 derivatives equals 20,000,000,000,000 trade price evaluations every day.

In this paper efficient Monte Carlo frameworks for pricing and measuring counterparty risk are developed.

CHAPTER 2

PRICING MODELS AND UNDERLYING DERIVITIVES

2.1 CVA

This section will develop a general framework for pricing CVA as well as describe the underlying assumptions made in modelling CVA. Gregory (2020) presents a more general overview and derivations of the formulas presented here. The notation and framework used in this paper is adapted from Hofer and Karlsson (2017) and tailored to the equations used throughout.
The CVA of a derivatives portfolio P can mathematicallly be presented as follows:

$$\text{CVA}(P) = E\left[(LGD)\frac{V_P^+(\tau)}{B(\tau)}\mathbf{I}(\tau < T)\right]$$

where LGD denotes the the loss given deafult, $V_P^+(\tau) = \max(V_P(\tau), 0)$, is the positive value of the derivative portfolio P at the time of the counterparty default, $\tau > 0$, $B(t)$ is the numeraire under the simulated measure (risk-free discount factor) , and T is the maximum maturity of all the derivatives that lie within the portfolio P.

Assuming that the default of the counterparty has occurred at time $\tau = t$, the CVA can defined as the integrated expected exposure discounted to time zero multiplied by the default probability, that is,

$$\text{CVA}(P) = (LGD)E\left[\int_0^T \frac{V_P(t)^+}{B(t)}dF(t)\right]$$

where $F(t)$ is the counterparty's cumulative default probability at time t. The default probability is usually modeled as

$$F(t) = 1 - \exp\left(-\int_0^t \gamma(s)ds\right)$$

where the (stochastic) function $\gamma(t)$ is called the hazard rate, the instantaneous credit spread or the conditional default probability.

Gregory (2020) gives a commonly used approximation for the risk-neutral default probabilities between any two sequential dates as:

$$PD\left(t_{i-1}, t_i\right) \approx \exp\left(-\frac{s_{t_{i-1}} t_{i-1}}{LGD}\right) - \exp\left(-\frac{s_t t_i}{LGD}\right)$$

as well as the derivation of approximate risk-neutral default probabilities (Gregory Appendix 12A, 2020). This approach is useful for pricing since it defines the price in terms of hedging instruments and supports the accounting standard's exit price notion.

The typical CVA formula is calculated under the assumption that credit risk and market risk factors are uncorrelated and that default probabilities are deterministic. The formula can then be written as:

$$\text{CVA}(P) = LGD \int_t^T E\left[\frac{V_P(t)^+}{B(t)}\right] dF(t)$$

It should be noted that this does not account for WWR and only applies to unilateral CVA.

To approximate CVA using Monte Carlo, a discrete time grid , $[t = t_0, t_1, \ldots, t_m = T]$ is defined for m points which are used to estimate the the exposure at these time points. An approximation of the CVA can then be written as

$$\text{CVA}_P \approx LGD \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{M} \frac{V_P\left(t_j, X_i\right)^+}{B\left(t_j, X_i\right)} \left[F\left(t_j\right) - F\left(t_{j-1}\right)\right]$$

where X_i is random variable vector that could contain all necessary simulated market parameters used to price CVA along the pathway, and the total number of simulated pathways is N. Gregory (2020) argues that this approximation holds well when M is large enough. Hofer and Karlsson (2017) states that exposure grids of $100 - 200$ points distributed over 50 years are utilized in practice, with shorter intervals between points for short maturities and longer intervals for long maturities.

2.1.1 Stand alone, Incremental and Marginal CVA

The formulas described previously are for standard stand alone CVA calculations at the portfolio level. Suppose the party has other positions with this counterparty in a netting set; CVA will then be expressed at the netting set level. When a new position is priced, the value of the counterparty risk of the position should be calculated and charged to the counterparty.

The following question then needs to be answered, "How much does this new trade change the CVA?". This question is answered by calculating the incremental CVA, which is the CVA with the trade included in the netting set minus the CVA without the trade in the netting set. This approach has a draw back though, incremental CVAs are not additive and depend on the order in which the transactions are added to the netted set. If the CVA of the netting set needs to be decomposed to asses each trade's contribution, marginal CVA should be used. Marginal CVA indicates how much each trade contributes to the total CVA. Marginal CVAs are additive but there there is a caveat. As new trades are added to the netting set, the marginal CVAs of all the existing trades changes.

Marginal CVA is still good for assessing each trade's contribution to the overall nettingset CVA, but this means we cannot charge the counterparty the marginal CVA. Why? Since the marginal CVA of a trade depends on /changes the marginal CVA of all the other trades in the nettingset, the marginal CVA of the newly added trade may only account for that part of the counterparty risk and the remaining fraction should never be charged to the client. Throughout this paper we will focus on stand alone CVA, but the methods developed are also applicable to incremental CVA. Since incremental CVA is used to price new trades, efficient Monte Carlo are only necessary for this purpose. Marginal CVA could be seen as a risk practice measurement where valuation of CVA in real time is not as important.

2.1.2 CVA as a runnning spread

Although CVA is frequently conceived of and characterized as a stand-alone charge, in many circumstances, such as on top of a swap contract's fixed spread, it is more accurate to quote it as a running spread. This makes comparing various trading quotations for a client a lot easier. When a spread is added to a contract, such as a swap, the problem becomes nonlinear since the spread will affect the CVA. The right number should be determined recursively, ensuring that the CVA charge accurately offsets the CVA incorporated into the contract.

2.2 UNDERLYING STOCHASTIC PROCESSES

Let a risk factor be represent by the random variables, $\{X_t; t \geq 0\}$, which follows a stochastic process defined on a given filtered probability space, $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0t\infty}, P)$. Ghamami and Zhang (2014) gives the following three continuous time stochastic processes that are widely used in finance and economics:

1. A Gauss-Markov process specified by

$$\mathrm{d}X_t = (f_t + g_t X_t)\,\mathrm{d}t + \sigma_t\ \mathrm{d}W_t$$

   with the deterministic functions f, g and σ of time and W the usual one dimensional Brownian motion.

2. A geometric Brownian motion (GBM)

$$\mathrm{d}X_t = \mu X_t\ \mathrm{d}t + \sigma \mathrm{d}W_t$$

   with given constants μ and σ.

3. A square-root diffusion specified as

$$\mathrm{d}X_t = \alpha\,(b - X_t)\,\mathrm{d}t + \sigma\sqrt{X_t}\ \mathrm{d}W_t$$

   in which α and b are positive.

The GBM process is for example often used to model stock prices, while square-root diffusion processes such as the Cox-Ingersoll-Ross interest rate model are frequently used to model interest rates. Ghamami and Zhang (2014) gives the following explanation of how simulation can be achieved via these stochastic processes.

Equate, $X_i \equiv X_{t_i}$, for the finite-dimensional distribution of X on a time grid $t_1, \ldots, t_n$. Assume that $X = (X_1, \ldots, X_n)$ can be sampled from precisely, in the sense that the distribution of the simulated X process at times $t_1, \ldots, t_n$ is precisely that of the process X; examples include Brownian motion processes and the square-root diffusion described above, both of which simulate positively correlated normal random variables. Let $\tilde{X} = \left(\tilde{X}_1, \ldots, \tilde{X}_n\right)$ denote a random vector for which $\tilde{X} \overset{d}{\neq} X$ but $\tilde{X}_i \overset{\mathrm{d}}{=} X_i$ for all $i = 1, \ldots, n$ and $\operatorname{cov}\left(\tilde{X}_i, \tilde{X}_j\right) = 0$ for all $i \neq j$. Therefore the simulation of $\tilde{X}_1, \ldots, \tilde{X}_n$ can be accomplished by generating n uncorrelated or simply independent normal random variables.

2.2.1 Brownian Bridges

Glasserman (2004) defines a Brownian bridge as a continuous-time stochastic process, $B(t)$, whose probability distribution is the conditional probability distribution of a standard Wiener process $W(t)$ subject to the condition (when standardized) that $W(T) = 0$, allowing the process to be pinned to the same value at both $t = 0$ and $t = T$. To be more precise

$$B_t := (W_t \mid W_T = 0), t \in [0, T]$$

Renzitti *et al.* (2020) gives following formula for a Brownian bridge discretization:

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

where $s < t < u$ and $\xi_t \sim \mathcal{N}(0, 1)$. The Brownian bridge should be constructed through simulation in such a way that each independent Brownian motion explains the most amount of variation that remains after the previous Brownian motion. Assume we require Brownian movements in M equal time steps, with M being a power of two. Renzitti *et al.* (2020) gives the following algorithm that could be used to construct the Brownian paths:

1. Set initial value of $W_0 =$ eqaul to 0.
2. Simulate $\hat{W}_M$ as $\hat{W}_M = \sqrt{T}\xi_0$, where ξ_0 is an vector of normal iid rvs.
3. The Brownian bridge discretization formula 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) for a full description of the process.

Each each random draw adds finer and finer features to the path, increasing the reliance on fewer and earlier indexed uniform variables and lowering the effective dimension for payoffs that are essentially determined by the Brownian path's overall shape.

CHAPTER 3

MONTE CARLO SIMULATION

3.1 STANDARD MONTE CARLO CVA

In many finance applications, pricing of financial instruments depends on underlying risk factors. Theses risk factors are written as a vector or random variables, X. These random variables are often standard normal, which is denoted by Z. For example, when using GBM for asset pricing, payoffs typically can be written in the form $X = h(Z_1, \ldots, Z_m)$, where m is the time points. The following example is used to explain the simple Monte Carlo approach to estimate CVA when we have one underlying risk factor and a GBM process is used to simulate the path of the underlying asset.

For each simulation i of a GBM process we generate $Z_{i,1}, Z_{i,2}, ..., Z_{i,T}$, independent normal random variables, where T is a time greater than the longest outstanding contract[1]. The GBM is then used to calculate the path of the asset X_i. A time grid $t_1, t_2, ..., t_n$, where $n < T$, is then specified at which points to value $V_P(t_j, X_i^+)$. The finer the time grid, the more valuation will be needed, and thus the computing times of the estimator will increase along with the accuracy of the estimator. A time grid should therefore be chosen to optimise the trade off between accuracy and computing time. This process is repeated N times, so that there are N simulated paths, and valuations at each time points in the grid for say, n time points. There is therefore $N \times n$ valuations required. Ghamami and Zhang (2014) looks at a optimisation technique to find the optimal value of grid points and numbers of simulations. The valuations of the instruments in a derivative portfolio P, $V_P(t_j, X_i^+)$ for all $j = \{1, \ldots, n\}$ and $i = \{1, \ldots, n\}$, are then simply plugged into Monte Carlo approximation for CVA derived in section 2.1. It should be noted that this process could also be followed for other credit risk measures such as EPE and EEPE, as well as other valuations adjustments such as FVA and KVA. An example of the simulated Browninan paths is displayed in Figure 3.1.

[1]The increments between the simulation here are daily, however a finer time grid could be used.

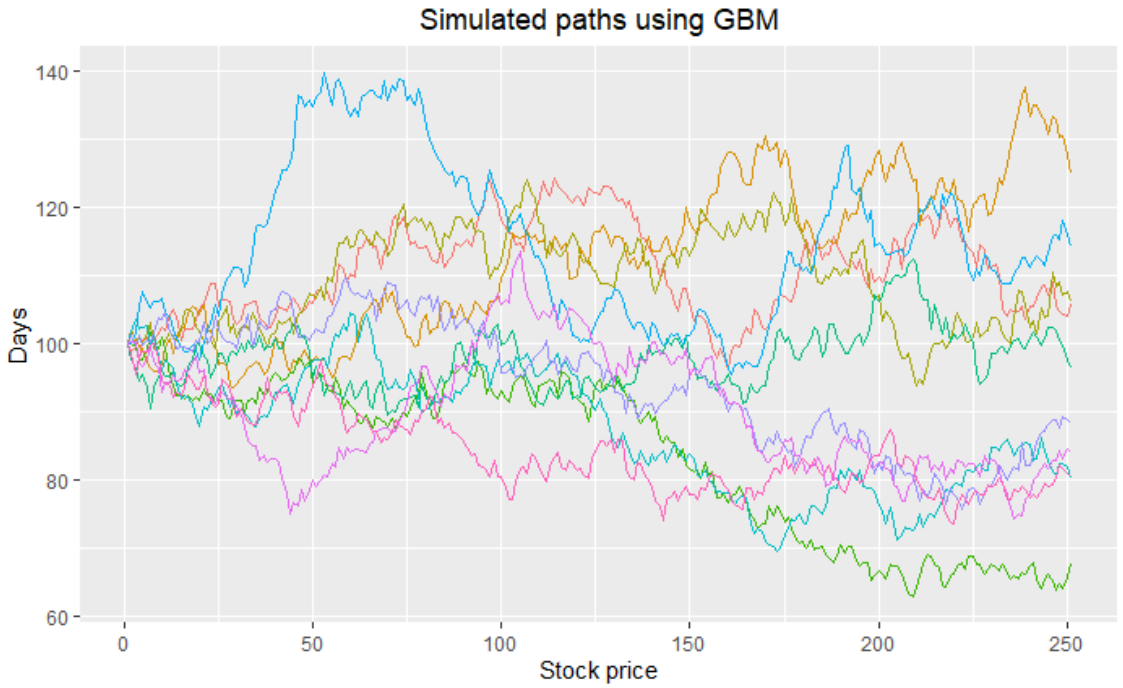

Figure 3.1: Simulated paths of a stock price using geometric Brownian motion.

3.2 PDS VS DJS SAMPLING

There are two main sampling methods that are used when counterparty risk measures is estimated. Path dependant simulation (PDS) is when the underlying risk factors are sampled from finite dimensional distributions. Direct jump to simulation date is a sampling technique that uses the notion of marginal matching to compute the risk factors. Pykhtin and Zhu (2006) show that under certain conditions and assumptions these sampling methods can be considered equivalent. Ghamami and Zhang (2014) however, has shown that DJS produces lower mean square error estimates for certain credit risk measure estimates and that this approach should be preferred to PDS. In this paper PDS is used due to its simplicity and intuitive understanding. The methods proposed could also be used when DJS sampling is used.

3.3 EFFICIENT MONTE CARLO

Monte Carlo integration can be applied to estimate functions of the type $E[g(X)]$. Efficient Monte Carlo refers to methods applied to find more efficient ways of computing Monte Carlo estimates of

$E[g(X)]$. Methods that the reduce the variance of the estimates can be seen as more efficient since less simulation are required in order to obtain the same level of accuracy as for standard Monte Carlo. The metric used in this paper to evaluate the efficiency of the proposed methodologies is therefore variance reduction[2]. Rizzo (2019) gives the following equation to calculate the percentage variance reduction of an estimator in relation to another.

Let $\theta = E[g(X)]$ the sample mean of the estimator. Suppose $\hat{\theta}_1$ and $\hat{\theta}_2$ are estimators of the parameter θ, and $\text{Var}\left(\hat{\theta}_2\right) < \text{Var}\left(\hat{\theta}_1\right)$, then the percentage variance reduction achieved by using $\hat{\theta}_2$ rather than $\hat{\theta}_1$ is :

$$100\left(\frac{\text{Var}\left(\hat{\theta}_1\right) - \text{Var}\left(\hat{\theta}_2\right)}{\text{Var}\left(\hat{\theta}_1\right)}\right)$$

3.4 ANTITHETIC VARIABLE

The first approach this paper will look at is the antithetic variable approach. Suppose there is a path that uses the following variables $\{\varepsilon_1, \ldots, \varepsilon_T\}$, the antithetic path then consists of taking the negative of these variables, $\{-\varepsilon_1, \ldots, -\varepsilon_T\}$. The idea behind antithetic variables is explained as follows. Suppose that the estimate of

$$\theta = \text{E}(g(X)) = \text{E}(Y)$$

is required. To achieve this two samples are generated, Y_1 and Y_2. An unbiased estimate of θ is given by

$$\hat{\theta} = \frac{Y_1 + Y_2}{2}$$

The formula for the variance of the estimate is then

$$\text{Var}(\hat{\theta}) = \frac{\text{Var}\left(Y_1\right) + \text{Var}\left(Y_2\right) + 2\,\text{Cov}\left(Y_1, Y_2\right)}{4}$$

so that the variance is reduced if $\text{Cov}\left(Y_1, Y_2\right)$ is negative. If Z is a normally distributed random variable then ,$-Z$, is also unit normal, and the correlation coefficient between Z and $-Z$ is $\rho = -1$.

[2]Computing time could also be used, however this varies depeding on the implementation of the algorithms.

This is due to the following proposition stated without proof.

Proposition 3.1

Suppose there is a function f that generates $X = g(Z_1, \ldots, Z_n)$, which is monotone in each variable. Then $X_1 = g(Z_1, \ldots, Z_n)$ and $X_2 = g(-Z_1, \ldots, -Z_n)$ are negatively correlated, i.e. Z_i $\mathrm{Cov}(X_1, X_2) < 0$, where$Z_i$ is iid $N(0, 1)$.

An example of the antithetic paths is given in Figure 3.2.

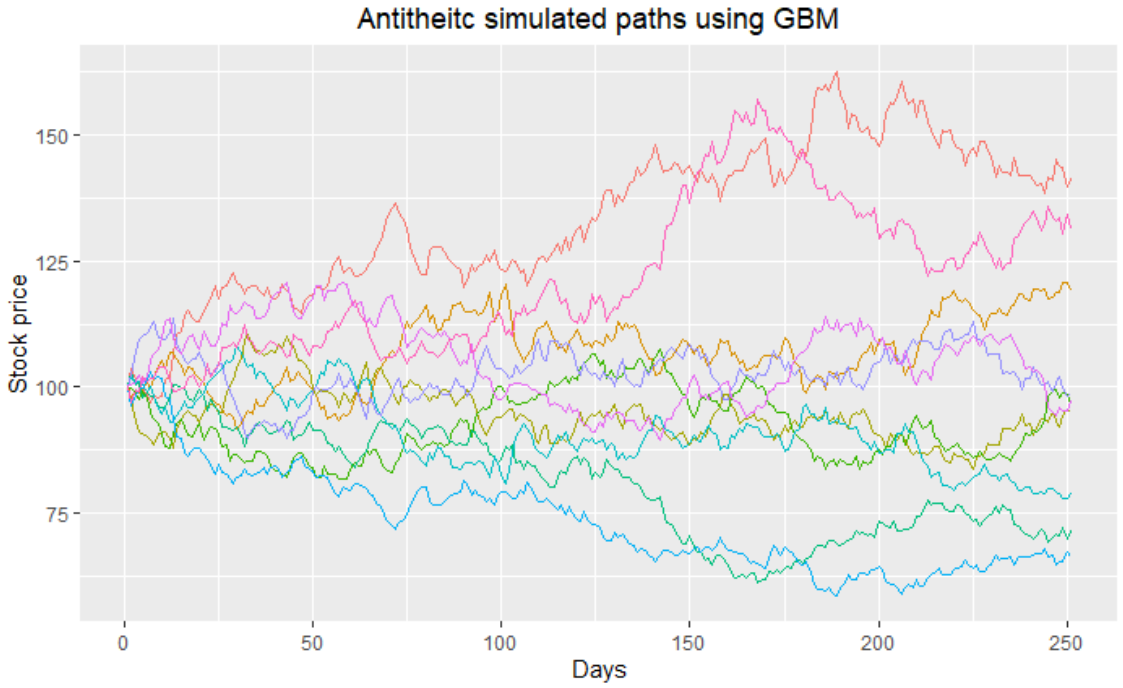

Figure 3.2: Antithetic simulations of a stock price using geometric Brownian motion

Suppose for example the exposure for an European call option is to be calculated at time points $0 = t_0 < t_1 < t_2 < \cdots < t_m = T$. The exposure at time point t_j is then

$$V_P(t_j, X_i)^+ = f(S(t_j))$$

where $f()$ is some function to determine the value of the option, for example using the Black-Scholes formula, and $S(t_j)$ is the stock price at time t_j. The following algorithm could be used to

construct the antithetic pairs which is adapted from (Ramström, 2017) to fit to the CVA procedure.

Antithetic algorithm for calculating CVA of a European call option

1. Simulate m iid $N(0,1)$ randon variables, $Z_1, \ldots, Z_m$. Then set

$$Q_i = e^{\sigma\sqrt{t_i - t_{i-1}}Z_i + \mu(t_i - t_{i-1})}, i \in \{1, 2, \ldots, m\}$$

2. Recursively set

$$\begin{aligned} S(t_1) &= S(0)Q_1 \\ S(t_2) &= S(t_1)Q_2 = S_0Q_1 \times Q_2 \\ &\vdots \\ S(t_m) &= S(t_{m-1})Q_m = S_0Q_1 \times Q_2 \times \cdots \times Q_m \end{aligned}$$

Set

$$\text{CVA}_{P,1} = LGD \sum_{j=1}^{M} \frac{V_P(t_j, S(t_j))^+}{B(t_j, S(t_j)} \left[F(t_j) - F(t_{j-1})\right]$$

3. Now reset the Q_i in (1) by using $-Z_1, \ldots, -Z_m$ in place of $Z_1, \ldots, Z_m$, that is, set

$$Q_i = e^{-\sigma\sqrt{t_i - t_{i-1}} - Z_i + \mu(t_i - t_{i-1})}, i \in \{1, 2, \ldots, m\}$$

4. Recursively set

$$\begin{aligned} S(t_1) &= S(0)Q_1 \\ S(t_2) &= S(t_1)Q_2 = S_0Q_1 \times Q_2 \\ &\vdots \\ S(t_m) &= S(t_{m-1})Q_m = S_0Q_1 \times Q_2 \times \cdots \times Q_m \end{aligned}$$

Set

$$\text{CVA}_{P,2} = LGD \sum_{j=1}^{M} \frac{V_P(t_j, S(t_j))^+}{B(t_j, S(t_j)} \left[F(t_j) - F(t_{j-1})\right]$$

5. Set

$$X = \frac{CVA_{P,1} + CVA_{P,2}}{2}$$

Denote $X_1 = X$, and generate a second independent copy by simulating independent $Z_1, \ldots, Z_m$ and set $X_2 = X$. Repeat the procedure N times to obtain the N desired estimates for CVA [3].

3.5 STRATIFIED SAMPLING

Stratified random sampling is a method of sampling that involves the division of a population into smaller sub-groups known as strata. The strata can be divided into strata that have similar characteristics. A proportional amount of samples is then drawn from each strata. The idea behind stratified sampling is that by sampling from strata that are divided by different characteristics a more 'fair" sample can be drawn that represents the whole populations and the different characteristics and it reduces the risk of drawing a biased sample.

A mathematical proof is given by (Glasserman, 2004) and presented by (Ramström, 2017) which shows why the variance is always smaller for stratified sampling than for standard MC.

The option price can be written as the discounted payoff of the expected value, Y, the stratification variable, X, and $\{A\}_{i=1}^{m}$ which is disjointed subsets of R^d. These subsets are chosen $P\left(X \in \bigcup_i A_i\right)$ and the expected value of the discounted option price can be written as

$$E[Y] = \sum_{i=1}^{M} E\left[Y \mid X \in A_i\right] \cdot P\left(X \in A_i\right) = p_i \cdot E\left[Y \mid X \in A_i\right]$$

Define:

$$\mu_i = E\left[Y_{ij}\right] = E\left[Y \mid X \in A_i\right]$$
$$\sigma_i^2 = \operatorname{Var}\left[Y_{ij}\right] = \operatorname{Var}\left[Y \mid X \in A_i\right]$$

The variance for simple MC is :

$$\operatorname{Var}[\hat{Y}] = \operatorname{Var}\left[\frac{1}{m}\sum_{i=1}^{m} Y_i\right] = \frac{1}{n^2} \cdot n \operatorname{Var}\left[Y_1\right] = \frac{1}{n}\left(E\left[Y_1^2\right] - E\left[Y_1\right]^2\right) = \frac{1}{n}\left(E\left[Y_i^2\right] - \mu^2\right)$$

[3] When antithetic variables are used, often N/2 paths are simulated, to get N/2 antithetic paths. The total number of paths is then N. This is he method also used in the paper

Where the mean μ is equal to

$$\mu = E[Y_1] = \sum_{i=1}^{m} p_i E[Y_i \mid X \in A_i] = \sum_{i=1}^{m} p_i \cdot \mu_i$$

and therefore the expectation of Y_1^2 is

$$E\left[Y_i^2\right] = \sum_{i=1}^{m} p_i E\left[Y_i^2 \mid X \in A_i\right] = \sum_{i=1}^{m} p_i \left(\sigma_i^2 + \mu_i^2\right)$$

since

$$\mathrm{Var}[Y_1 \mid X \in A_i] = E\left[Y_1^2 \mid X \in A_i\right] - E[Y_1 \mid X \in A_i]^2 \Longleftrightarrow \sigma^2 = E\left[Y_1^2 \mid X \in A_i\right] - \mu^2$$

The simple MC variance is then

$$\begin{gathered}\mathrm{Var}[\hat{Y}] = \frac{1}{n}\left(\sum_{i=1}^{m} p_i \left(\sigma_i^2 + \mu_i^2\right) - \mu^2\right) \\ = \frac{1}{n}\left(\sum_{i=1}^{m} p_i \sigma_i^2\right) + \frac{1}{n}\left(\sum_{i=1}^{m} p_i \mu_i^2\right) - \frac{1}{n}\left(\sum_{i=1}^{m} p_i \mu_i\right)^2\end{gathered}$$

The variance when using stratification is

$$\mathrm{Var}\left[\hat{Y}_S\right] = \mathrm{Var}\left[\sum_{i=1}^{m} p_i \cdot \frac{1}{n_i} \sum_{i=1}^{n_i} Y_{ij}\right] = \sum_{i=1}^{m} p_i^2 \cdot \mathrm{Var}\left[\frac{1}{n_i} \sum_{i=1}^{n_i} Y_{ij}\right] = \sum_{i=1}^{m} p_i^2 \frac{\sigma_i^2}{n_i} = \frac{\sigma^2(q)}{n}$$

where

$$\sigma^2(q) = \sum_{i=1}^{m} \frac{p_i^2}{q_i} \cdot \sigma_i^2, q_i = \frac{n_i}{n}$$

With proportional stratification $p_i = q_i$ we get

$$\sigma^2(q) = \sum_{i=1}^{m} \frac{p_i^2}{q_i} \sigma_i^2 = \sum_{i=1}^{m} p_i \sigma_i^2$$

using proportional stratification the equation can be written as

$$\mathrm{Var}\left[\hat{Y}_S\right] = \frac{1}{n} \sum_{i=1}^{m} p_i \sigma_i^2$$

A comparison of these two variance calculations gives

$$\text{Var}[\hat{Y}] - \text{Var}\left[\hat{Y}_S\right] = \frac{1}{n}\left(\sum_{i=1}^{m} p_i \mu_i^2\right) - \frac{1}{n}\left(\sum_{i=1}^{m} p_i \mu_i\right)^2$$

which by Jensen's inequality results that

$$\sum_{i=1}^{m} p_i \mu_i^2 \geq \left(\sum_{i=1}^{m} p_i \mu_i\right)^2$$

which mathematically proves that stratified sampling will always produce of lower variance of the estimate then for simple MC.

A simplistic way to implement this approach is given by the following algorithm. This algorithm will be used in the numerical results.

Stratified MC sudo algorithm

1. Simulate N paths and calculate the N estimates for the CVA.
2. Sort the N CVA estimates in ascending order.
3. Stratify the samples into n equally proportional strata.
4. Randomly draw m samples from each n strata and compute the mean of each samples from the strata
5. Computes the means of the n stratified means
6. Repeat steps (5) and (6) for N/m/n times.

An interesting question is then what is the optimal strata and the amount of samples drawn from each strata but is behind the scope of this paper. Another way to utilise stratified sampling is by sampling from strata that are divided by interval in the underlying distribution of drawn samples. Rizzo (2019) gives this process in more detail and shows that the variance can be reduced significantly.

3.6 QUASI MONTE CARLO

Xiao (2015) describes the quasi-Monte Carlo(QMC) method as a method used for numerical integration and solving some other problems using low-discrepancy sequences (LDS) (also called quasi-random sequences or sub-random sequences). The underlying idea of QMC is similar to that of classical Monte Carlo simulation with the exception that the pseudo-random numbers (PRN) are replaced with carefully selected low-discrepancy sequences that are more evenly distributed.

Renzitti *et al.* (2020) states that the QMC method is an acceleration technique that is commonly used to price single trade payoffs. QMC has been shown to result in convergence rates between $\mathcal{O}(n^{\frac{-1}{2}})$ and $\mathcal{O}(n^{-1})$. This convergence rate is dependant on the complexity of the derivative, and to be more precise, dependant on the interacting terms between the uniform random variables. Derivatives that are heavily dependant on these interaction terms often result in convergence rates closer to classical Monte Carlo, $\mathcal{O}(n^{\frac{-1}{2}})$, whereas derivatives that are not so dependant usually results in convergence rates closer to $\mathcal{O}(n^{-1})$.

In this paper the following methodology will be used. At maturity of the derivative a value for the underlying stochastic process will be simulated by using the inverse transformation of the distribution on a *Sobol* [4] sequence for N values. That is $W(T_1), \ldots, W(T_N)$ will be simulated, where $W(T_i) = \Phi^{-1}(S_{T_i}), i = 1, ..N$ and $(S_1, \ldots, S_N)$ is the Sobol sequence required. Brownian bridges could then be constructed to obtain the earlier values $W(t_1), \ldots, W(t_{m-1})$. Then the formula for simple MC could be used on the simulated values to obtain the QMC. The idea behind this method is that end values for the underlying values could be used as a proxy for the expected exposure and therefore also CVA. Then by applying Quasi-Monte Carlo these values will be more evenly distributed than for the normal pesudo random methodology.

[4] Sobol sequences use a base of two to form successively finer uniform partitions of the unit interval and then reorder the coordinates in each dimension

3.7 MULTI-LEVEL MONTE CARLO

Multi-level Monte Carlo (MLMC) is typically used in financial settings where the underlying asset can be modelled as a stochastic differential equation which is used to calculate the price of the derivative.Hofer and Karlsson (2017) stat that MLMC methods can greatly reduce the computational cost of standard Monte Carlo methods by taking most samples with a low accuracy and corresponding low cost, and only very few samples are taken at high accuracy and corresponding high cost. Hofer and Karlsson (2017) investigate the use of MLMC to reduce the number of simulation required to find the optimal CVA running spread for an interest rate swap .The authors show that significant reduction in computation time of between 65% and 85% compared to standard MC can be achieved by using MLMC.

CHAPTER 4

NUMERICAL RESULTS

4.1 PRACTICAL IMPLEMENTATION

This section will look at implementation of the proposed methodologies described in the previous section. The CVA for a European call option is calculated using standard Monte Carlo (SMC) and compared to the efficient Monte Carlo methods discussed earlier. To simplify the results risk mitigates such as netting and collateral have been excluded. The only risk factor simulated was that of an stock price that followed a Geometric Brownian motion. Risk neutral default probabilities are calculated using the approximation given by (Gregory, 2020). The default values and assumptions for the calculation of the estimators is displayed Table 4.1. At each grid point the Black Scholes formula is used to price the European call option. It should be noted that this methodology is also applicable to Asian options as well as exotic options.

Credit Spread	150 bp	Stock Pice	100
Interest Rate	5%	Strike price	100
Time to Maturity (in Years)	1	Drift	0
Increments	1/250	Volatility (constant)	0.25
Number of Simulations	10000	LGD	0.6
Grid points	50		

Table 4.1: Default values used to estimate CVA

The variance reduction techniques that will be compared to SMC are: Antithetic Monte Carlo (AMC), stratified Monte Carlo (SFMC) and Quasi Monte Carlo (QMC). The following results displayed in this section shows the CVA estimates as certain parameters are changed to check the robustness of the estimators. Table 4.2 displays the CVA estimator for an out of the money, at the money and in the money call option with strikes prices K=105, K= 100 and K=95 respectively.

	Out of the money			**At the money**		
	Estimate	***Variance***	***Var Red***	***Estimate***	***Variance***	***Var Red***
Simple MC	-0.1220	0.0129	-	-0.1619	0.0183	-
Antithetic MC	-0.1224	0.0029	77.5901	-0.1622	0.0034	81.4930
Stratified MC	-0.1229	0.0001	99.5516	-0.1629	0.0001	99.5723
Quasi MC	-0.1033	0.0065	49.5405	-0.1423	0.0097	46.9048
	In the money					
	Estimate	***Variance***	***Variance Reduction***			
Simple MC	-0.1995	0.0224	-			
Antithetic MC	-0.1996	0.0033	85.1148			
Stratified MC	-0.2006	0.0001	99.5902			
Quasi MC	-0.1808	0.0124	44.8472			

Table 4.2: Comparison of estimators for CVA of OTM, ATM an ITM European Option.

The results clearly display that the stratified estimator achieves the most significant variance reduction with more than 99 % variance reduction for all three types of options and with little variation. The antithetic estimator results in variance reduction of between 77 % and 85 %, with the variance decreasing modestly as the option shifts more in the money. The variation reduction for the QMC estimator varies between 45 % and 50% with the variance increasing as the option strike price increases. The antithetic and stratified slightly overestimates the the SMC estimator but is almost exactly the same, while the QMC underestimates it by a larger margin.

Table 4.3 displays the results when the time to maturity , T, of the option is increased to two years. The CVA naturally increases as the T increases, and both the stratified and antithetic estimators variance decreases as T is increased while the QMC estimator variance relative increases.

	T=1			**T=2**		
	Estimate	***Variance***	***Var Red***	***Estimate***	***Variance***	***Var Red***
Simple MC	-0.1619	0.0183	-	-0.2822	0.0465	0.0000
Antithetic MC	-0.1622	0.0034	81.4930	-0.2824	0.0073	84.3179
Stratified MC	-0.1629	0.0001	99.5723	-0.2825	0.0002	99.6561
Quasi MC	-0.1423	0.0097	46.9048	-0.2646	0.0295	36.4576

Table 4.3: Comparison of estimators for CVA for different maturities of underlying

Table 4.4 displays the results when the number of simulations , N, is decreased from 10 000 to 1000. As expected the variance increases for all estimators although not significantly. The estimates are also further of from each other.

	N= 10 000			**N = 1000**		
	Estimate	***Variance***	***Var Red***	***Estimate***	***Variance***	***Var Red***
Simple MC	-0.1619	0.0183	-	-0.1689	0.0209	0.0000
Antithetic MC	-0.1622	0.0034	81.4930	-0.1702	0.0043	79.6634
Stratified MC	-0.1629	0.0001	99.5723	-0.1695	0.0000	99.8949
Quasi MC	-0.1423	0.0097	46.9048	-0.1435	0.0100	52.4161

Table 4.4: Comparison of estimators for CVA for different number of simulations

As a last result, the estimators are a looked at when CVA increases quite significantly. The credit spread of the counterparty is increased to 400 basis points and therefore the default probabilities will increase and subsequently the CVA. Table 4.5 displays these results and the only thing to take note here is that the QMC estimate is inaccurate compared to the other estimator.

	CS = 400		
	Estimate	***Variance***	***Var Red***
Simple MC	-0.4169	0.1184	0.0000
Antithetic MC	-0.4177	0.0217	81.6776
Stratified MC	-0.4195	0.0005	99.5678
Quasi MC	-0.3669	0.0626	47.1251

Table 4.5: CVA estimator for higher credit spread

From the combined results of the simulation for the different estimators and parameters values there are a couple points to take away. Firstly, the AMC and SFMC estimators produce accurate results in comparison with the SMC estimator, while the QMC estimator tends to underestimate CVA a lot more. The reason is probably due to some slight error in the construction of the Brownian bridges and that the values of the GBM at maturity is only approximation of the expected exposure over the life of the option and not necessarily accurate. Secondly, the SFMC estimator achieves significant variance reduction consistently, while the AMC achives modest variance reduction that varies slightly depending on the parameters used to value the derivative and that is used in the calculation of CVA. The SFMC could also be used in combination with both of the other methods which coould lead to even more signifacnt variance reduction

CHAPTER 5

CONCLUSION

The paper proposed mainly three methodologies to produce efficient Monte Carlo estimators of CVA. The theoretical explanations held well against the practical implementation, which validated the proposed methods. The stratified Monte Carlo method produced the greatest variance reduction consistently. This methods could also be used in combination with other Monte Carlo methods. Antithetic sampling also produced significant variance reduction when Monte Carlo simulation was applied. The quasi Monte Carlo method produced modest variance reduction, while the estimates were slightly off.

The methods proposed in this paper could also be applied to other financial derivatives such as interest rate swaps. Other counterparty default risk metrics such as EE and EEPE could also be estimated with these methods. Since these methods are mainly applied to the underlying stochastic processes, other xVA estimates could also make use of these methods.

REFERENCES

Caflisch, R.E., Morokoff, W.J. and Owen, A.B. (1997). *Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension*, vol. 24. Department of Mathematics, University of California, Los Angeles.

Ghamami, S. and Zhang, B. (2014). Efficient monte carlo counterparty credit risk pricing and measurement. *Journal of Credit Risk*, vol. 10, no. 3.

Glasserman, P. (2004). *Monte Carlo methods in financial engineering*, vol. 53. Springer.

Gregory, J. (2020). *The XVA Challenge: Counterparty Risk, Funding, Collateral, Capital and Initial Margin*. John Wiley & Sons.

Hofer, M. and Karlsson, P. (2017). Efficient calibration for cva using multi-level monte carlo. *Available at SSRN 2776932*.

Hull, J. and White, A. (2012). Cva and wrong-way risk. *Financial Analysts Journal*, vol. 68, no. 5, pp. 58–69.

Pykhtin, M. and Zhu, S.H. (2006). Measuring counterparty credit risk for trading products under basel ii. *RISK Books*.

Ramström, A. (2017). Pricing of european and asian options with monte carlo simulations: Variance reduction and low-discrepancy techniques.

Renzitti, S., Bastani, P. and Sivorot, S. (2020). Accelerating cva and cva sensitivities using quasi-monte carlo methods. *Wilmott*, vol. 2020, no. 108, pp. 78–93.

Rizzo, M.L. (2019). *Statistical computing with R*. CRC Press.

Xiao, T. (2015). An accurate solution for credit value adjustment (cva) and wrong way risk. *The Journal of Fixed Income*, vol. 25, no. 1, pp. 84–95.