HKUST MSc. Financial Mathematics
MAFS 5220 Quantitative and Statistical Risk Analysis

Unilateral Credit Valuation Adjustment for FX Forward
Methodology and Validation Report

17 May 2015

Prepared by Group 5

Team Member:
Kenric Cheong (CHEONG, Siu Sing) - 20279891
Ken Tsui (TSUI, Hin Kan) - 08109319
Zhou Lei (XU, Zhou Lei) - 20279889

Table of contents

1. Executive Summary

In this report, we explained our motivation of selecting credit valuation adjustment of FX forward as our project aim.

We then explained the theory and methodology behind and illustrated how we constructed the model using QuantLib. Major classes and program structure are also explained in detailed.

Afterwards, we validated our model by testing the assumptions, well designed test cases, and also benchmarking. Model limitation and restriction are also well noted.

Finally, we concluded that our model can be used in normal market circumstances and suggested area s of improvement for future enhancement.

We hope that this report can document our model construction and model validation process so it reduces the model risk and also enable others to replicate our work. We would also like to thank Professor Chak Wong and Mr. Felix Lee for their insight and guidance.

2. Motivation

Since the 2008 global financial crisis, counterparty risk management has been a huge topic.

Particularly, Basel III and International Financial Reporting Standards both requires OTC derivative to account for credit loss provision in order to reflect true value.

What is CVA?

Literally speaking, Credit Valuation adjustment (CVA) means the day zero's expectation of expected discounted positive exposure at default until the maturity of a derivative contract after considering the default probability and loss given default.

Why CVA?

- Risk based pricing is the key for obtaining a competitive position in market while also pricing in the counterparty risk as a reserve.

- Risk management can use CVA limit as a one of the risk metrics to monitor. CVA VaR is another type.

- Financial and regulatory reporting requires CVA to be calculated to reflect the fair value.

Why FX Forward?

- Globally and in Hong Kong, FX derivative is far more common than the interest rate derivative. More importantly, its turnover experienced very high growth from 2010 to 2013. In Hong Kong SAR, the FX market is almost the 10 times of the interest rate derivatives.

TABLE 1

Average daily turnover of the global FX and OTC interest rate derivatives markets (by geographical distribution)

US$ billion

	Foreign exchange market		Interest rate derivatives		Total	
Economy	April 2013	April 2010	April 2013	April 2010	April 2013	April 2010
United Kingdom	2,726	1,854	1,348	1,235	4,074	3,088
United States	1,263	904	628	642	1,891	1,546
Singapore	383	266	37	35	420	301
Japan	374	312	67	90	441	402
Hong Kong SAR	275	238	28	18	303	256
Switzerland	216	249	33	75	249	324
France	190	152	202	193	392	345
Australia	182	192	66	41	248	233
Netherlands	112	18	29	61	141	80
Germany	111	109	101	48	212	157
Denmark	103	120	59	16	162	137
Canada	65	62	34	42	99	104
Russia	61	42	0.2	–	61	42
Luxembourg	51	33	0.4	2	52	36
Korea	48	44	8	11	55	55
China	44	20	13	2	57	21
Sweden	44	45	17	18	61	63
Spain	43	29	14	31	57	60
Mexico	32	17	2	1	35	18
India	31	27	3	3	35	31
Others	318	310	68	84	387	394
Total turnover [1 & 3]	**6,671**	**5,043**	**2,759**	**2,649**	**9,430**	**7,692**
Estimated global turnover [2]	**5,345**	**3,971**	**2,343**	**2,054**	**7,688**	**6,025**

Notes:
1. Data adjusted to exclude double counting of figures reported by local inter-dealer.
2. Data adjusted to exclude double counting of figures reported by local and cross-border inter-dealer.
3. Figures may not add up to total due to rounding.

Source: The Bank for International Settlements

Source: HKMA (2013), The foreign exchange and derivatives markets in Hong Kong

- FX swap, another type of common instruments, is effectively a pair of a spot FX transaction, and a forward FX transaction. FX swap and FX forward together has constituted more than 60% of the market. Given this, we want to write a program, which can cater for the majority of CVA calculation.

TABLE 2

Average daily turnover of the Hong Kong FX market (by instrument)

US$ billion

	Turnover in Hong Kong			Global turnover	
	April 2013	April 2010	Change (%)	April 2013	April 2010
Spot	51.2	43.8	16.9	2,459	1,829
Outright forwards	37.3	32.0	16.5	816	558
Foreign exchange swaps	174.1	147.0	18.4	2,931	2,352
Currency swaps	2.5	7.0	-64.1	68	57
OTC options and other OTC products	9.5	7.7	23.3	397	246
Total foreign exchange transactions	274.6	237.6	15.6	6,671	5,043

Notes:
1. Average daily turnover has been adjusted to exclude double counting of figures reported by local inter-dealer.
2. Figures may not add up to total due to rounding.
3. Other OTC products have not been adjusted to exclude double counting of figures reported by local inter-dealer.

Source: HKMA (2013), The foreign exchange and derivatives markets in Hong Kong

- FX forward is usually structured as investment product, for example, ratio forward and target redemption forward. By subsequently altering the payoff function, this program can be catered to any FX product.

Why deal level CVA?

- Although it is not as accurate as portfolio incremental CVA, its computation is much faster and requires much data.

- It is always more conservative because it ignores the portfolio diversification effect

Why C++ and Quantlib?

- Quantlib is one of the most comprehensive and well tested quantitative finance library in the world

- CVA computation requires heavy computation power and therefore C++ works well.

- Program in C++ can also be interfaced with other system afterwards

3. Theory and Methodology

3.1. CVA Formula

Equation of CVA

$$\mathbb{E}_0[(1-REC)D(0,\tau)1_{\{\tau<T\}}(\mathbb{E}_\tau[\Pi(\tau,T)])^+]$$

Equation of Independent based CVA:

The independent based CVA assumes that the Exposure at default is independent of the hazard rate.

The conditional expectation of Exposure at Default is therefore equal to its unconditional expectation.

This is a bold assumption worth caution, but it simplifies the calculation power.

After discretization, it equals to:

$$(1-REC)\sum_{i=1}^{n}\mathbb{Q}\{\tau\,\epsilon(t_{i-1},t_i]\}\mathbb{E}_0[D(0,\tau)(\mathbb{E}_{t_i}[\Pi(t_i,T)])^+]$$

Parameter required:

1. Expected Exposure at Default time (EAD) at each time step at day zero expectation

2. Stochastic Discounting Factor at each time step at day zero expectation

3. Expected default probability of each time step at day zero expectation

4. Recovery Rate

3.2. Exposure at Default (EAD)

Exposure at Default - FX Forward

Fair Value of Fair Value at time t

= Notional Amount in terms of foreign currency x (S_t x $DF_{\text{Foreign Currency}}$ – K x $DF_{\text{Domestic Currency}}$)

where $DF_{\text{Foreign Currency}}$ = exp(-zero $\text{rate}_{\text{Foreign Currency}}$ x (T-t)),

$DF_{\text{Domestic Currency}}$ = exp(-zero $\text{rate}_{\text{Domestic Currency}}$ x (T-t)),

T is the maturity time of FX forward

S_t and K are quoted as the price in domestic currency per foreign currency.

Path Generation

Given the payoff function, in order to form an expectation, we have to utilize Monte Carlo Simulation by generating the paths from default time until the maturity at each time step.

Geometric Brownian Motion (GBM) is assumed to be the exchange rate stochastic process.

Stochastic Differential Equation:

$dS_t = \mu S_t dt + \sigma S_t dW_t$, where W_t is a Wiener Process.

Dynamics of S_t:

$$S_t = S_0 \exp((\mu - \frac{\sigma^2}{2})t + \sigma W_t)$$

In Monte Carlo simulation, a random number is generated to calculate W_t at every time step and at every path.

There are two parameters in the Monte Carlo simulation:

- Drift term of the Geometric Brownian Motion
- Volatility term of the Geometric Brownian Motion

The drift term and volatility term are designed to be user input.

Since these two terms are directly observable in the market (e.g. FX vol from FX option), we have set this as user inputs.

3.3. Stochastic Discounting Factor

Stochastic Discounting Factor – Vasicek Model

The interest rate is assumed to follow the Vasicek model (an one factor model).

Vasicek interest rate model is a short rate model. The interest rate process is as follows:

$$dr_t = \alpha(\gamma - r_t)dt + \sigma dz_t$$

There are two key functions of the Vasicek interest rate model:

- Simulate short rates
- Calculate the stochastic discount factor

Monte Carlo simulation with Vasicek interest rate model can be used to simulate short rates.

$$r_{t+\Delta t} = r_t e^{-\lambda\Delta t} + \mu\left(1 - e^{-\lambda\Delta t}\right) + \sigma\varepsilon\sqrt{\frac{1 - e^{-2\lambda\Delta t}}{2\lambda}}$$

In program, we use OrnsteinUhlenbeckProcess in Quantlib to generate the sample paths.

Before performing the simulation, the Vasicek interest rate model needs to be calibrated.

Calibration

There are three parameters in the Vasicek model (i.e. λ the mean reversion rate, μ the mean, σ the volatility). They can be estimated with maximum likelihood estimator as discussed in Herrala's paper.

Historical short rates (e.g. 1-month LIBOR) and the log-likelihood function are input into the Excel. Then set the Excel's solver to maximize the log-likelihood function by changing the values of λ, μ and σ with Newton-Rhapson optimization.

Although Vasicek interest rate model is an one factor model, it has a closed form solution of the price of a zero-coupon bond as mentioned in Emile van Elen's paper.

$$\mathrm{P}(\mathrm{t},\mathrm{T}) = \exp\left(-\left[\left(\lambda - \frac{\sigma^2}{2\mu^2}\right)(\mathrm{T}-\mathrm{t}) + \left(r_t - \lambda + \frac{\sigma^2}{\mu^2}\right)\frac{1 - e^{-\mu(T-t)}}{\mu} - \frac{\sigma^2}{2\mu^2}\frac{1 - e^{-2\mu(T-t)}}{2\mu}\right]\right)$$

The price of a zero-coupon is a function of short rate and some constant terms.

Emile shows how to use Vasicek interest rate model to calculate the future price of a zero-coupon bond, and the future zero-coupon bond is the stochastic discount factor that is needed in the calculation of unilateral CVA of a FX forward contract.

Calibration Algorithm

We calibrate the Vasicek Model (an Ornstein-Uhlenbeck process) by a time series of short rates.

The Vasicek Model parameters has a closed-form solution using Maximum Likelihood function according to a paper on the SITMO website.

The closed-form solution for the parameters are as follow:

Mean:

$$\mu = \frac{S_y S_{xx} - S_x S_{xy}}{n(S_{xx} - S_{xy}) - (S_x^2 - S_x S_y)}$$

Mean reversion rate:

$$\lambda = -\frac{1}{\delta} \ln \frac{S_{xy} - \mu S_x - \mu S_y + n\mu^2}{S_{xx} - 2\mu S_x + n\mu^2}$$

, where $\delta = \Delta t$, i.e. time increment

Variance:

$$\begin{aligned} \hat{\sigma}^2 =& \frac{1}{n}[S_{yy} - 2\alpha S_{xy} + \alpha^2 S_{xx} \\ & - 2\mu(1-\alpha)(S_y - \alpha S_x) + n\mu^2(1-\alpha)^2] \\ \sigma^2 =& \hat{\sigma}^2 \frac{2\lambda}{1-\alpha^2} \end{aligned}$$

With $\alpha = e^{-\lambda \Delta t}$, $S_x = \sum_{i=1}^{n} r_{i-1}$, $S_y = \sum_{i=1}^{n} r_i$, $S_{xx} = \sum_{i=1}^{n} r_{i-1}^2$, $S_{xy} = \sum_{i=1}^{n} r_{i-1} r_i$, $S_{yy} = \sum_{i=1}^{n} r_i^2$.

Path Generation

After calibrating the μ, λ and σ, random number with stand normal distribution is generated to simulate short rate r_t for each time step in each path.

3.4. Hazard Rate

Hazard Rate - Default Modelling

The hazard rate is assumed to be deterministic.

Piecewise Constant Intensity is adopted to calculate the probability of default due to the following reasons:

- It is easy to implement by means of discretization.

- It directly extracts information from the market (i.e. CDS quotes).

- It is better than a constant intensity method as the Piecewise Constant Intensity method considers the term structure of CDS.

- However, the hazard rate has zero volatility, which limits the model ability to have CVA simulation.

Since we measure the default probability at day zero and that we set the hazard rate to be deterministic, the hazard rate is **only** calibrated at the beginning with the day zero's observation of CDS spread and zero rate. The stochastic interest rate will not change the hazard rates afterwards.

Piecewise Constant Intensity

Hazard rate is the conditional default probability given that it does not default in the previous period.

$$Q(\tau \in [t, t+dt] | \tau > t, F) = \gamma(t)dt$$

For a Piecewise Constant Intensity, it becomes:

$$Q(\tau \in [t_{i-1}, t_i] | \tau > t_{i-1}, F) = \gamma dt$$

Survival Function:

$$Q(\tau > t) = \exp\left(-\Gamma(t)\right) = \exp(-\gamma(t)t)$$

Probability of Default Function:

$$Q(\tau < t) = 1 - \exp\left(-\Gamma(t)\right) = 1 - \exp(-\gamma(t)t)$$

Calibration - Linking Hazard Rate to Running CDS Quotes

The CDS premium leg and protection leg both depend on survival function and probability of default function. By equating the premium leg, which depends on the running CDS spread, to protection leg, the hazard rate can be backed out.

Premium Leg $= -\mathrm{R} \int_{T_a}^{T_b} P(0,t) \left(t - T_{\beta(t)-1}\right) d_t Q(\tau \geq t)) + R \sum_{i=a+1}^{b} P(0,T_i) \alpha_i Q(\tau \geq T_i)$ (Brigo3.3.1)

$= -\mathrm{R} \int_{T_a}^{T_b} P(0,t) \left(t - T_{\beta(t)-1}\right) d_t \exp(-\Gamma(t)) + R \sum_{i=a+1}^{b} P(0,T_i) \alpha_i \exp(-\Gamma(T_i))$

$= -\mathrm{R} \int_{T_a}^{T_b} P(0,t) \left(t - T_{\beta(t)-1}\right)(-\gamma)\exp(-\Gamma(t))\, dt + \mathrm{R} \sum_{i=a+1}^{b} P(0,T_i) \alpha_i \exp(-\Gamma(T_i))$

$= \mathrm{R} \int_{T_a}^{T_b} P(0,t) \left(t - T_{\beta(t)-1}\right)(\gamma)\exp(-\Gamma(t))\, dt + \mathrm{R} \sum_{i=a+1}^{b} P(0,T_i) \alpha_i \exp(-\Gamma(T_i))$

After discretization,

Premium Leg $\sim R\ \sum_{i=a+1}^{b} P(0,T_i)(T_i - T_{i-1})(\gamma_i)\exp\left(-\Gamma(T_i)\right)(T_i - T_{i-1})$ + $\mathrm{R}\sum_{i=a+1}^{b} P(0,T_i)\alpha_i \exp(-\Gamma(T_i))$

= $\mathrm{R}\sum_{i=a+1}^{b} P(0,T_i)(\alpha_i)(\gamma_i)\exp\left(-\Gamma(T_i)\right)(T_i - T_{i-1})$+ $\mathrm{R}\sum_{i=a+1}^{b} P(0,T_i)\alpha_i \exp(-\Gamma(T_i))$

= $\mathrm{R}\sum_{i=a+1}^{b} P(0,T_i)(\alpha_i)(\gamma_i)\exp\left(-\Gamma(T_i)\right)\alpha_i$+ $\mathrm{R}\sum_{i=a+1}^{b} P(0,T_i)\alpha_i \exp(-\Gamma(T_i))$

= $\mathrm{R}\sum_{i=a+1}^{b} P(0,T_i)(\alpha_i^2)(\gamma_i)\exp\left(-\Gamma(T_i)\right)$+ $\mathrm{R}\sum_{i=a+1}^{b} P(0,T_i)\alpha_i \exp(-\Gamma(T_i))$

Protection Leg=$-$ LGD$\int_{T_a}^{T_b} P(0,t)\, d_t Q(\tau \geq t)$ (Brigo3.3.1)

=$-\ \int_{T_a}^{T_b} P(0,t)\, d_t \exp(-\Gamma(t))$

=$-\int_{T_a}^{T_b} P(0,t)\,(-\gamma)\exp(-\Gamma(t))\, dt$

=$\int_{T_a}^{T_b} P(0,t)\,\gamma \exp(-\Gamma(t))\, dt$

After discretization,

Protection Leg $\sim \sum_{i=a+1}^{b} P(0,T_i)(\gamma_i)\exp\left(-\Gamma(T_i)\right)(T_i - T_{i-1}) = \sum_{i=a+1}^{b} P(0,T_i)(\gamma_i)\exp\left(-\Gamma(T_i)\right)\alpha_i$

Calibration Algorithm

1. Obtain the first R (CDS quote of nearest maturity) and input into the formula
2. Set Premium Leg equals to Protection leg, that is

   $\sum_{i=a+1}^{b} P(0,T_i)(\alpha_i^2)(\gamma_i)\exp\left(-\Gamma(T_i)\right)$+ $\sum_{i=a+1}^{b} P(0,T_i)\alpha_i \exp(-\Gamma(T_i))$ -$\sum_{i=a+1}^{b} P(0,T_i)(\gamma_i)\exp\left(-\Gamma(T_i)\right)\alpha_i$=0
3. For quarterly payment, set $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4$, $\alpha_i = 0.25$ for all i
4. γ_1,γ_2,γ_3,γ_4 are backed out
5. Obtain the Second R and input it with γ_1,γ_2, γ_3, γ_4. Set the formula to zero to get γ_5,γ_6, γ_7,γ_8.
6. Repeat until solving all the hazard rate for R.

3.5. Recovery Rate

The recovery rate is assumed to be deterministic at 40% due to the following reasons:

- There is limited recovery data for building a statistical model.

- The market practitioners tend to adopt 40% recovery rate in CDS context.

- However, the recovery rate has zero volatility, which limits the model ability to have CVA simulation.

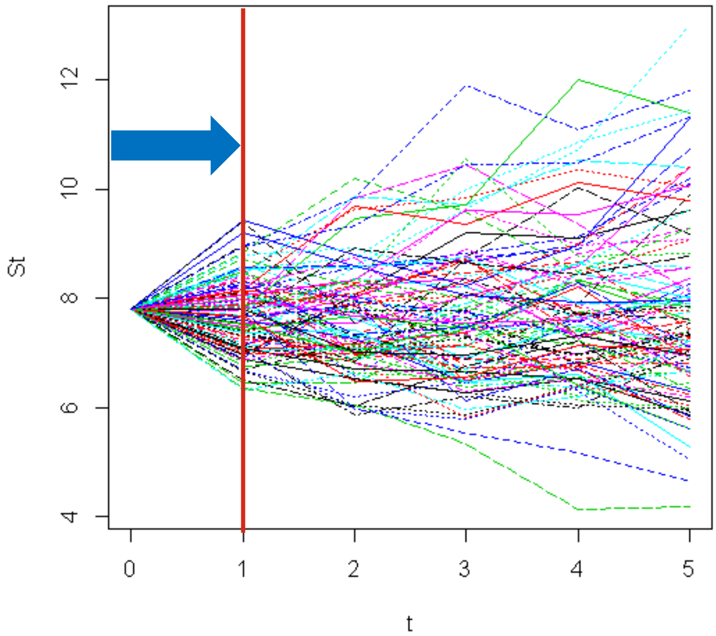
3.6. The Calculation of CVA

Recall the formula: recovery rate is constant across time while hazard rate is piecewise constant across different timestep. The last item is the day zero's expectation of thediscounted expectation at that time step of positve exposure from each time step to maturity.

$$(1 - REC)\sum_{i=1}^{n} \mathbb{Q}\{\tau\,\epsilon(t_{i-1}, t_i]\}\mathbb{E}_0[D(0,\tau)(\mathbb{E}_{t_i}[\Pi(t_i, T)])^+]$$

To calculate the last item, the expected positive exposure from each time step to maturity is fully revaluated at each time step. For example, in t=1, we simulate the exposure and get the expected value. Then we go to t=2, take the expected value of the simulated S2 as starting point and simulate the exposure at t=2. Likewise, we further go to t=3,4,5 until the maturity.

The expected exposure at each time step is discounted at the stochastic interest rate of that time step and then multiplied by the hazard rate at that time step and the constant recovery rate.



The example has an original maturity of t=6.

After simulation from t1 to t6 to calculate EAD at t=1, we then shift one day and calculate the EAD at t2 [simulation from t2 to t6].

The expected value of S2 will be used to start the simulation from t2 to t6 to calculate EAD at t2.

4. Model Construction

4.1. Input Specification

- Valuation date
- Maturity date
- Position of the FX forward contract – long or short
- FX spot rate at the valuation date
- Contract rate / strike price of the FX forward contract
- Notional amount of the FX forward contract
- Domestic bond price, coupon for bootstrapping the zero curve (Excel File)
- Foreign bond price, coupon for bootstrapping the zero curve (Excel File)
- Time series of interest rate for calibrating the discount factor (Excel File)
- CDS Spread [1Y,2Y,3Y,4Y,5Y,7Y,10Y]
- Recovery Rate

4.2. Major Classes Used

Major classes used in different component of the models:

Components	**Classes Used**
EAD Modelling	GeometricBrownianMotionProcess MersenneTwisterUniformRng BoxMullerGaussianRng RandomSequenceGenerator PathGenerator
Stochastic Discount Factor Modelling	OrnsteinUhlenbeckProcess
Hazard Rate Modelling	InterpolatedZeroCurve SpreadCdsHelper PiecewiseDefaultCurve hazardRate
LGD	NA

We aimed to describe the function of the selected classes literally, and avoid complex technicality.

For readers who want to explore the classes more in depth, please always refer to the Quantlib documentation.

GeometricBrownianMotionProcess

This class is inherited to the class <StochasticProcess>. This class defines the stochastic process.

3 arguments are required:

S0 - Initial value of the simulation

Mu - drift of the stochastic process

Sigma - volatility of the stochastic process

This class is then the input of the class <PathGenerator>.

```
const boost::shared_ptr<StochasticProcess>& gbm = boost::shared_ptr<StochasticProcess>(new
GeometricBrownianMotionProcess(S0, mu, sigma));
```

MersenneTwisterUniformRng

This class is uniform number generator. In our model, this is assigned with the seed, which determines the sequence of random variable.

```
BigInteger seed_fx = SeedGenerator::instance().get();
MersenneTwisterUniformRng mersenneRng_fx(seed_fx);
```

BoxMullerGaussianRng

This is to converting uniform random number to Gaussian distribution (normal distribution) random variables.

```
typedef BoxMullerGaussianRng<MersenneTwisterUniformRng> MersenneBoxMuller;
```

RandomSequenceGenerator

This is to form a sequence of random number after generating normal random variables.

The first argument is the dimensionality while the second is the random variables that you want to use in the sequence.

```
RandomSequenceGenerator<MersenneBoxMuller> gsg_fx(1, boxMullerRng_fx);
```

PathGenerator

This is to generate paths of defined stochastic process (gbm in this example) and the random sequence generated (gsg_fx in this example). User has to time length and time steps as well.

```
PathGenerator<RandomSequenceGenerator<MersenneBoxMuller>> fxPathGenerator_t0(gbm, dt_1d,
1, gsg_fx, false);

const Path& fxpath_t0 = fxPathGenerator_t0.next().value;
```

OrnsteinUhlenbeckProcess

This class is inherited to the class <StochasticProcess>. This class defines the stochastic process.

4 arguments are required:

lamda_mll - speed λ of the Vasicek model.

sigma_mll - volatility σ of the Vasicek model.

r0 - Initial value of the simulation.

mu_mll - level μ of Vasicek model.

This class is then the input of the class <PathGenerator>.

```
const boost::shared_ptr<StochasticProcess>& ir = boost::shared_ptr<StochasticProcess>(new
OrnsteinUhlenbeckProcess(lamda_mll, sigma_mll, r0, mu_mll));
```

InterpolatedZeroCurve

This class can interpolate the zero curve. It can inherit to YieldTermStructure. Depending on the method of interpolation, user can replace <Linear> by <Cubic>, <LogLinear> with underlying parameter,etc. In our cases, the dates are the calender dates while the rates are the zero rates at different tenors. Daycounter in this example is Actual 365 Fixed().

```
Handle<YieldTermStructure> irTermStructure(boost::shared_ptr<InterpolatedZeroCurve<Linear>
>(new InterpolatedZeroCurve<Linear>(dates, rates, dayCounter)));
```

SpreadCdsHelper

This class can inherit to the class Defaultprobabilityhelper. It is a CDS hazard rate bootstrap helper.

Key arguments include:

quoted_spreads[k]- a vector of CDS quotes

tenors[k] - a vector of CDS tenor

Quarterly - Frequency of payment

recovery_rate - Recovery rate

irTermStructure - Class of <YieldTermStructure>: interest rate term structure

settlesAccrual - This indicates whether there is a on default payment.

paysAtDefaultTime – This indicates whether the payment of premium leg and protection leg should be paid at default time or payment date.

```
SpreadCdsHelper(Handle<Quote>(boost::shared_ptr<Quote>(
                        new SimpleQuote(quoted_spreads[k]))),
                        tenors[k],
                        0,
                        calendar,
                        Quarterly,
                        Following,
                        DateGeneration::TwentiethIMM,
                        dayCounter,
                        recovery_rate,
                        irTermStructure, 1, 1)
```

PiecewiseDefaultCurve

This class can inherit to <hazardRateStructure>. This class sets the default curve by inputting the instruments which are passed to DefaultProbabilityHelper (so it is used together with SpreadCdsHelper in this example).

Key arguments include the valuation date and instrument that’s passed to DefaultProbabilityHelper.

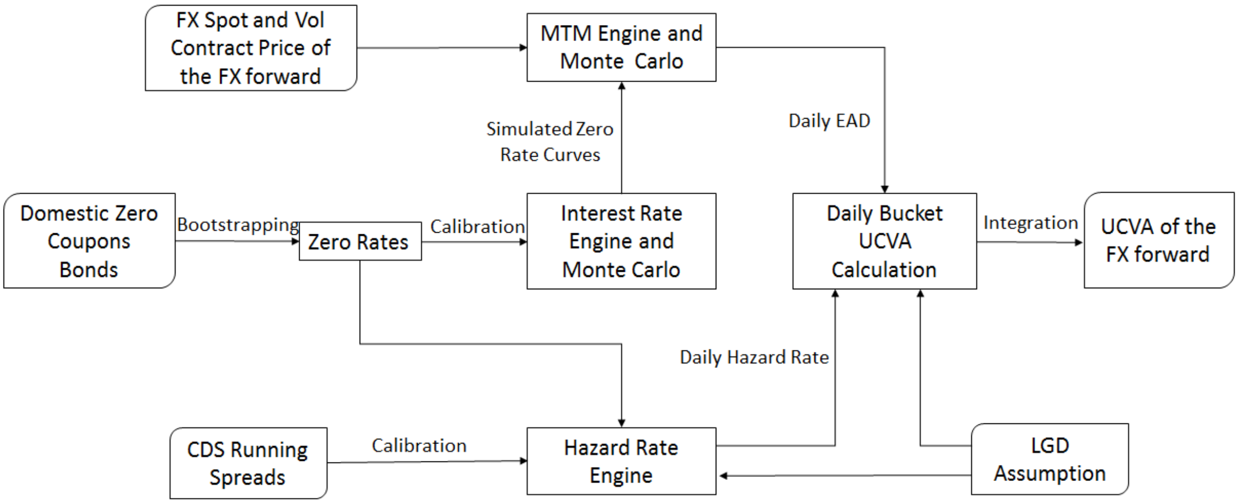
```
boost::shared_ptr<PiecewiseDefaultCurve<HazardRate, BackwardFlat> >
       hazardRateStructure(new PiecewiseDefaultCurve<HazardRate, BackwardFlat>(t0,
instruments, dayCounter));
```

hazardRate

This class is member of <hazardRateStructure>. It is to extract hazard rate by inputting the date.

```
PD[i] = hazardRateStructure->hazardRate(todaysDate) / 365;
```

4.3. Schematic Diagram



4.4. Program Structure

The calculation of CVA involves multiple Monte Carlo Simulations. This section will talk about the simulation algorithm. In the program, we have set everyday a timestep.

1. At the valuation date t0, multiple runs of simulations are performed to simulate FX rates and interest rates.

2. The average of the simulated FX rates and the average of the simulated interest rates from step 1 are taken as the expected FX rate and interest rate respectively at t1.

3. Then another multiple runs of simulations are performed with the expected FX rate and interest rate from step 2. Another simulated FX rates and interest rates from these simulations are used to calculate the EAD at t1 perspective (conditional to the information available at t1). In addition, the average of the simulated interest rates from step 2 are taken as the expected FX rate and interest rate respectively at t2.

4. Step 3 is repeated at each future day until to one day before the maturity.

5. In other words, the simulation at t is based on the simulation result at t-1. The result at t is conditional on the information available up to time t-1.

6. With this algorithm, the EAD at any future time T can be simulated with the information available up to time T. This follows the definition of the conditional expectation with the information available at time T.

7. Multiple the EAD with PD and LGD at any time t gives the CVA component at time t.

8. Summing the CVA component of each date up to one day before maturity gives the CVA.

9. Steps 1 to 8 are repeated multiple times in order to obtain a CVA with acceptable variance.

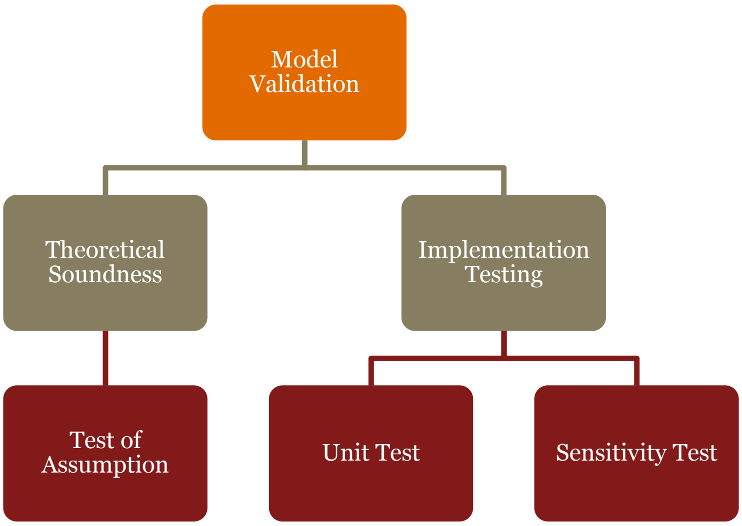
The program code is attached here.

Group 5.cpp

5. Validation

5.1. Validation Strategy

We first tested the model assumption and concluded the model limitation. Then, we tested the model implementation by well-designed test cases to check whether the C++ implementation is as what we expected. First set of test cases are component based, which tests individual component; the second set of test cases serve as overall sanity check of the direction of CVA shocked by parameters.



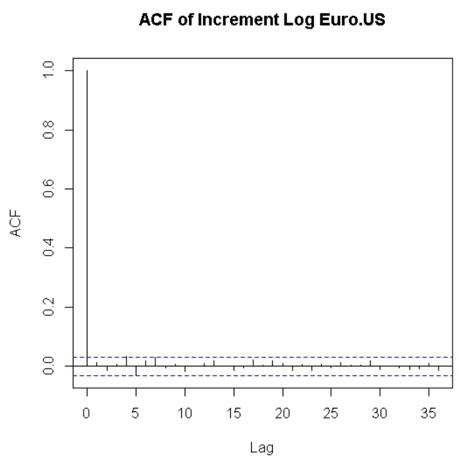
5.2. Theoretical Soundness

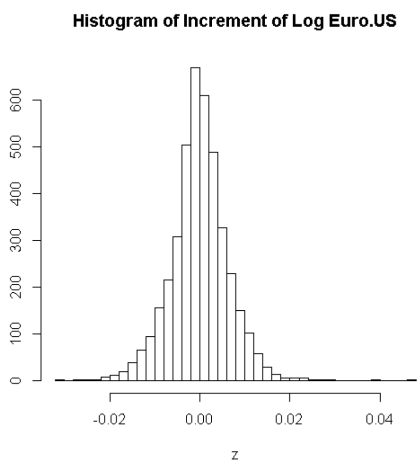
Test of Assumptions

Assumption: FX follows Geometric Brownian Motion (Increment is serially independent and normally distributed)

FX Rate is assumed to follow Geometric Brownian motion. In order to test whether FX exhibits independent increment, we obtained US/EUR from 1999/1 to 2015/5. We first took nature log of the FX rate and then took a first difference. The ACF (autocorrelation) graph of the increment is plotted. We observed that all lag are not significant. This implies that the logarithm of FX rate is a ARIMA(0,1,0) process, which is a random walk.

Also, we plotted a histogram for all the increment. It behaves like a normal distribution, with the mean around zero.






```
> jarque.bera.test(z)

        Jarque Bera Test

data:  z
X-squared = 812.1781, df = 2, p-value < 2.2e-16
```

However, when we performed Jarque Bera Test in R, the test implies there is excess skewness and kurtosis. Therefore the model does not perform well when skewness/ kurtosis of FX is high.

Assumption: Interest rate follows Vasicek Process

Vasicek model is a one factor model. It cannot capture the complex shift of term structure. In other words, it can model the "level" of yield curve but not the "twisting" yield curve.

There is no jump in the stochastic process. Therefore this model may not be reliable in stress period, in which there are spikes.

Therefore, we will impose a model restriction that this model cannot be used in stress period. On the other hand, in future enhancement, two factor model and jump will be considered.

Assumption: Recovery rate is deterministic (40%)

We have benchmarked the ISDA Standard CDS Model, which adopts a recovery rate of40% is used for senior unsecured while 20% is used for subordinate; 25% is used for emerging markets (both senior and subordinate). Therefore the 40% assumption is considered not unreasonable. Also, this assumption is both for hazard rate model and CVA formula, which sort of cancel out with each other.

Assumption: Independence of Exposure at Default and Hazard Rate

Our hazard rate is set to be deterministic and directly implied from the CDS (model independent). Therefore, we could not include a correlation between the EAD and the hazard rate. In the future enhancement, a stochastic intensity model will be considered.

5.3. Implementation – Test Cases

5.3.1. Unit Test

FX Forward

13 test cases have been prepared are used to test the program. The fair values of the FX forward are calculated for the 13 test cases, and the results are compared with the outcomes from the excel spread.

Below is the list of test cases:

1. Strike price / contract rate = 0
2. FX spot rate = 0
3. Notional amount = 0
4. FX spot rate is increased by 30%
5. FX spot rate is decreased by 30%
6. Notional amount is increased by 30%
7. Notional amount is decreased by 30%
8. Strike price / contract rate is increase by 30%
9. Strike price / contract rate is decreased by 30%
10. Zero rate curve of the domestic currency parallel shifts up by 200bps
11. Zero rate curve of the domestic currency parallel shifts down by 200bps
12. Zero rate curve of the foreign currency parallel shifts up by 200bps
13. Zero rate curve of the foreign currency parallel shifts down by 200bps

For the fair values, if the percentage difference between the value calculated by the program and the excel spread is within +/- 5%, the program output is deemed acceptable.

Below is the testing result of the program:

```
Test 1: K = 0 passes.
Test 2: S0 = 0 passes.
Test 3: Notional Amount = 0 passes.
Test 4: S0 is increased by 30% passes.
Test 5: S0 is decreased by 30% passes.
Test 6: Notional amount is increased by 30% passes.
Test 7: Notional amount is decreased by 30% passes.
Test 8: Contract rate is increased by 30% passes.
Test 9: Contract rate is decreased by 30% passes.
Test 10: Domestic zero rate curve shifts up by 200bps passes.
Test 11: Domestic zero rate curve shifts down by 200bps passes.
Test 12: Foreign zero rate curve shifts up by 200bps passes.
Test 13: Foreign zero rate curve shifts down by 200bps passes.
```

Attachment:

MAFS 5220_Group 5_Delta KKI_FX Fo

Vasicek Model

It is a known fact that the Vasicek model cannot fit very well into the real life interest rate. Nevertheless, it is still widely used due to its easy implementation. Therefore, we will just focus on testing the estimation algorithm.

Item	Quantlib	Excel	Difference
Mean reversion parameter (λ)	0.065316	0.065345	-0.05%
Equilibrium rate (μ)	0.242934	0.242915	0.01%
Volatility (σ)	0.004748	0.004748	0.00%

The difference is minimal.

```
C:\windows\system32\cmd.exe
The size of interest rate index: 253
The size of interest rate lag index: 253
Sx:58.8705
Sxx:13.7597
Sy:58.8832
Syy:13.766
Sxy:13.7626
maximum log-likelihood function method:
n2:252
lamda_mll:0.0653156
mu_mll:0.242934
sigma_mll:0.00474836
Press any key to continue . . .
```

Hazard Rate Model

For Test 1 – Test 7, the model passes the test if every piecewise hazard rate is within 5% of the expected result from the excel.

Test 1: CDS Spread parallel shift +/-50bps

Cds Spread		Interest rate	
1Y	117.5	1Y	1.00%
2Y	134.5	2Y	1.00%
3Y	150.5	3Y	1.00%
4Y	170	4Y	1.00%
5Y	187.57	5Y	1.00%
7Y	211.52	7Y	1.00%
10Y	228.66	10Y	1.00%

Recovery Rate
0.4

Maturity	Quanlib	Excel	Difference
1Y	0.0195589	0.019583	-0.12%
2Y	0.02536	0.025346	0.06%
3Y	0.0307239	0.030721	0.01%
4Y	0.0389219	0.038928	-0.02%
5Y	0.044382	0.044406	-0.05%
7Y	0.0471135	0.047172	-0.12%
10Y	0.0468032	0.046857	-0.11%

Cds Spread		Interest rate	
1Y	17.5	1Y	1.00%
2Y	34.5	2Y	1.00%
3Y	50.5	3Y	1.00%
4Y	70	4Y	1.00%
5Y	87.57	5Y	1.00%
7Y	111.52	7Y	1.00%
10Y	128.66	10Y	1.00%

Recovery Rate
0.4

Maturity	Quanlib	Excel	Difference
1Y	0.00291307	0.002917	-0.12%
2Y	0.00866513	0.008630	0.40%
3Y	0.0139364	0.013914	0.16%
4Y	0.0219207	0.021908	0.06%
5Y	0.0271816	0.027186	-0.02%
7Y	0.0297659	0.029801	-0.12%
10Y	0.0294186	0.029450	-0.11%

Variance <5%, no exception noted.

Test 2&3: CDS Curve Twist (tenor > 4Y:+/- 50bps ; tenor <4Y: -/+ 50bps)

Cds Spread		Interest rate	
1Y	17.5	1Y	1.00%
2Y	34.5	2Y	1.00%
3Y	50.5	3Y	1.00%
4Y	170	4Y	1.00%
5Y	187.57	5Y	1.00%
7Y	211.52	7Y	1.00%
10Y	228.66	10Y	1.00%

Recovery Rate
0.4

Maturity	Quanlib	Excel	Difference
1Y	0.00291307	0.002917	-0.12%
2Y	0.00866513	0.008630	0.40%
3Y	0.0139364	0.013914	0.16%
4Y	0.0934939	0.093957	-0.49%
5Y	0.0447674	0.044798	-0.07%
7Y	0.0474047	0.047468	-0.13%
10Y	0.0469681	0.047024	-0.12%

Cds Spread		Interest rate	
1Y	117.5	1Y	1.00%
2Y	134.5	2Y	1.00%
3Y	150.5	3Y	1.00%
4Y	70	4Y	1.00%
5Y	87.57	5Y	1.00%
7Y	111.52	7Y	1.00%
10Y	128.66	10Y	1.00%

Recovery Rate
0.4

Maturity	Quanlib	Excel	Difference
1Y	Error	0.019583	#VALUE!
2Y	Error	0.025346	#VALUE!
3Y	Error	0.030721	#VALUE!
4Y	Error	-0.030224	#VALUE!
5Y	Error	0.026854	#VALUE!
7Y	Error	0.029557	#VALUE!
10Y	Error	0.029318	#VALUE!

Negative hazard rate is counter-intuitive. Quantlib results in error due to the 4th step (4Y calibration). We will not redesign our code as CDSs in real world are quoted in a way to keep hazard positive. We have also revised our test.

Test 2&3Additional: CDS Curve Twist (tenor > 4Y: no change; tenor <4Y: + 50bps)

Cds Spread		Interest rate	
1Y	117.5	1Y	1.00%
2Y	134.5	2Y	1.00%
3Y	150.5	3Y	1.00%
4Y	120	4Y	1.00%
5Y	137.57	5Y	1.00%
7Y	161.52	7Y	1.00%
10Y	178.66	10Y	1.00%

Recovery Rate: 0.4

Maturity	Quanlib	Excel	Difference
1Y	0.0195589	0.019583	-0.12%
2Y	0.02536	0.025346	0.06%
3Y	0.0307239	0.030721	0.01%
4Y	0.00364684	0.003787	-3.70%
5Y	0.0357612	0.035616	0.41%
7Y	0.0382158	0.038348	-0.35%
10Y	0.0380277	0.038071	-0.11%

Variance <5%, no exception noted. There is a variance of -3.7% in 4Y. No further investigation will be conducted due to the rare nature of the CDS curve (U shape at 4Y).

Test 4: Recovery Rate +/- 0.1

Cds Spread		Interest rate	
1Y	67.5	1Y	1.00%
2Y	84.5	2Y	1.00%
3Y	100.5	3Y	1.00%
4Y	120	4Y	1.00%
5Y	137.57	5Y	1.00%
7Y	161.52	7Y	1.00%
10Y	178.66	10Y	1.00%

Recovery Rate: 0.5

Maturity	Quanlib	Excel	Difference
1Y	0.0134832	0.013500	-0.12%
2Y	0.020425	0.020396	0.14%
3Y	0.026832	0.026817	0.05%
4Y	0.0366168	0.036617	0.00%
5Y	0.0431463	0.043170	-0.05%
7Y	0.0464387	0.046501	-0.13%
10Y	0.0460979	0.046153	-0.12%

Cds Spread		Interest rate	
1Y	67.5	1Y	1.00%
2Y	84.5	2Y	1.00%
3Y	100.5	3Y	1.00%
4Y	120	4Y	1.00%
5Y	137.57	5Y	1.00%
7Y	161.52	7Y	1.00%
10Y	178.66	10Y	1.00%

Recovery Rate: 0.3

Maturity	Quanlib	Excel	Difference
1Y	0.00963091	0.009643	-0.12%
2Y	0.014577	0.014556	0.15%
3Y	0.0191213	0.019109	0.06%
4Y	0.0260167	0.026012	0.02%
5Y	0.0305605	0.030570	-0.03%
7Y	0.0327851	0.032822	-0.11%
10Y	0.0324749	0.032510	-0.11%

Variance <5%, no exception noted.

Test 5: Zero Curve parallel shift +/- 2%

Cds Spread		Interest rate	
1Y	67.5	1Y	-1.00%
2Y	84.5	2Y	-1.00%
3Y	100.5	3Y	-1.00%
4Y	120	4Y	-1.00%
5Y	137.57	5Y	-1.00%
7Y	161.52	7Y	-1.00%
10Y	178.66	10Y	-1.00%

Recovery Rate
0.4

Maturity	Quanlib	Excel	Difference
1Y	0.0112637	0.011250	0.12%
2Y	0.0169963	0.016930	0.39%
3Y	0.0222179	0.022151	0.30%
4Y	0.0300785	0.030001	0.26%
5Y	0.0353786	0.035149	0.65%
7Y	0.0376413	0.037671	-0.08%
10Y	0.037362	0.037309	0.14%

Cds Spread		Interest rate	
1Y	67.5	1Y	3.00%
2Y	84.5	2Y	3.00%
3Y	100.5	3Y	3.00%
4Y	120	4Y	3.00%
5Y	137.57	5Y	3.00%
7Y	161.52	7Y	3.00%
10Y	178.66	10Y	3.00%

Recovery Rate
0.4

Maturity	Quanlib	Excel	Difference
1Y	0.0112084	0.011250	-0.37%
2Y	0.0170298	0.017047	-0.10%
3Y	0.0224464	0.022488	-0.18%
4Y	0.030779	0.030851	-0.23%
5Y	0.0365422	0.036478	0.18%
7Y	0.03912	0.039361	-0.61%
10Y	0.0389436	0.039088	-0.37%

Variance <5%, no exception noted.

Test 6&7: Zero Curve Twist (tenor > 4Y:+/- 2% ; tenor <4Y: -/+ 2%)

Cds Spread		Interest rate	
1Y	117.5	1Y	-1.00%
2Y	134.5	2Y	-1.00%
3Y	150.5	3Y	-1.00%
4Y	120	4Y	3.00%
5Y	137.57	5Y	3.00%
7Y	161.52	7Y	3.00%
10Y	178.66	10Y	3.00%

Recovery Rate
0.4

Maturity	Quanlib	Excel	Difference
1Y	0.0112637	0.011250	0.12%
2Y	0.0169963	0.016930	0.39%
3Y	0.0222111	0.022151	0.27%
4Y	0.0304529	0.030976	-1.69%
5Y	0.0374072	0.037343	0.17%
7Y	0.0397743	0.040028	-0.63%
10Y	0.0393291	0.039478	-0.38%

Cds Spread		Interest rate	
1Y	117.5	1Y	3.00%
2Y	134.5	2Y	3.00%
3Y	150.5	3Y	3.00%
4Y	120	4Y	-1.00%
5Y	137.57	5Y	-1.00%
7Y	161.52	7Y	-1.00%
10Y	178.66	10Y	-1.00%

Recovery Rate
0.4

Maturity	Quanlib	Excel	Difference
1Y	0.0112084	0.011250	-0.37%
2Y	0.0170298	0.017047	-0.10%
3Y	0.0224532	0.022488	-0.15%
4Y	0.0303968	0.029871	1.76%
5Y	0.0346492	0.034423	0.66%
7Y	0.0371246	0.037147	-0.06%
10Y	0.0370891	0.037035	0.15%

Variance <5%, no exception noted. There is a variance of -1.69%/1.76% in 4Y. No further investigation will be conducted due to the rare nature of the jump in yield curve. Different tenors should have correlations

Attachment:

Piecewise Constant
Hazard Rate Calibrat

5.3.2. Sensitivity Test

20 test cases have been prepared to test the calculation of the fair value and the CVA by the program. For each test case, only one factor is shifted, other factors are remained unchanged. Taking test case 1 contract rate = 0 as an example, only the contract rate is changed to zero. Other factors including notional amount, FX spot rate, zero rates, CDS spreads, recovery rate, position etc. are kept unchanged.

The sanity check are performed on the CVA from each test case. Since the fair value is default free (i.e. probability of default is not considered in the fair value), test cases 14 – 20 should have no impact on the fair value. Those test cases shift the recovery rate and CDS spreads which have no impact on the default free fair value.

The formula used to calculate the percentage difference is as follows:

$$\text{Percentage Difference} = \frac{Fair\ Value\ from\ the\ Program}{Fair\ Value\ from\ the\ Excel\ Spreadsheet} - 1$$

For the sanity check of the CVA, the expectation of the movement of CVA (i.e. CVA either increase, decrease, no change or equal to 0) is determined for each test case. If the movement of CVA of the test case is the same as the expectation, the sanity check passes.

Below is the list of test cases and the expectation of the movement of CVA:

1. Strike price / contract rate = 0

    - For a long position in the FX forward, the CVA is expected to be increased when contract rate = 0. This is because contract rate = 0 increases the EAD, thus CVA is expected to be increased.

    - For a short position in the FX forward, the CVA is expected to be zero when contract rate = 0. This is because contract rate = 0 makes the EAD zero, thus CVA is expected to be zero.

2. FX spot rate = 0

    - For a long position in the FX forward, the CVA expected to be zero when FX spot rate = 0. This is because FX spot rate = 0 makes the EAD zero, thus CVA is expected to be zero.

    - For a short position in the FX forward, the CVA is expected to be increased when FX spot rate = 0. This is because FX spot rate = 0 increases the EAD, thus CVA is expected to be increased.

3. Notional amount = 0

    - When notional amount = 0, EAD = 0. Thus CVA = 0.

4. FX spot rate is increased by 30%

    - For a long position in the FX forward, the CVA is expected to be increased when FX spot rate is increased. This is because an increase in FX spot rate increases the EAD, thus CVA is expected to be increased.

    - For a short position in the FX forward, the CVA is expected to be decreased when FX spot rate is increased. This is because an increase in FX spot rate makes the EAD smaller, thus CVA is expected to be decreased.

5. FX spot rate is decreased by 30%

   - For a long position in the FX forward, the CVA is expected to be decreased when FX spot rate is decreased. This is because a decrease in FX spot rate makes the EAD smaller, thus CVA is expected to be decreased.

   - For a short position in the FX forward, the CVA is expected to be increased when FX spot rate is decreased. This is because a decrease in FX spot rate increases the EAD larger, thus CVA is expected to be increased.

6. Notional amount is increased by 30%

   - The CVA is expected to be increased when notional amount is increased. This is because an increase of the notional amount makes the EAD larger, thus CVA is expected to be increased.

7. Notional amount is decreased by 30%

   - The CVA is expected to be decreased when notional amount is decreased. This is because a decrease of the notional amount makes the EAD smaller, thus CVA is expected to be decreased.

8. Strike price / contract rate is increase by 30%

   - For a long position in the FX forward, the CVA is expected to be decreased when contract rate is increased. This is because an increase of the contract rate decreases the EAD, thus CVA is expected to be decreased.

   - For a short position in the FX forward, the CVA is expected to be increased when contract rate is increased. This is because an increase of the contract rate increases the EAD, thus CVA is expected to be increased.

9. Strike price / contract rate is decreased by 30%

   - For a long position in the FX forward, the CVA is expected to be increased when contract rate is decreased. This is because a decrease of the contract rate increases the EAD, thus CVA is expected to be increased.

   - For a short position in the FX forward, the CVA is expected to be decreased when contract rate is decreased. This is because a decrease of the contract rate decreases the EAD, thus CVA is expected to be decreased.

10. Zero rate curve of the domestic currency parallel shifts up by 200bps

    - An increase in the zero rate curve of the domestic currency can impact the interest rate forecast, PD estimation, EAD estimation etc. As a result, it is hard to determine the movement of the CVA and the CVA is not tested for this test case.

11. Zero rate curve of the domestic currency parallel shifts down by 200bps

    - A decrease in the zero rate curve of the domestic currency can impact the interest rate forecast, PD estimation, EAD estimation etc. As a result, it is hard to determine the movement of the CVA and the CVA is not tested for this test case.

12. Zero rate curve of the foreign currency parallel shifts up by 200bps

    - Assuming the drift term of the Geometric Brownian motion is not changed, any change of the zero rate curve of the foreign currency has no impact on the CVA. In other words, CVA is expected to be unchanged.

13. Zero rate curve of the foreign currency parallel shifts down by 200bps

    - Assuming the drift term of the Geometric Brownian motion is not changed, any change of the zero rate curve of the foreign currency has no impact on the CVA. In other words, CVA is expected to be unchanged.

14. Recovery Rate = 0 assuming there is no change to the default rate

    - Assuming there is no change the default rate, CVA is expected to be increased when the recovery rate is 0. This is because a recovery rate of 1 means LGD = 1, CVA is expected to be increased enormously.

15. Recovery Rate = 1 assuming there is no change to the default rate

    - Assuming there is no change the default rate, CVA is expected to be zero when the recovery rate is 1. This is because a recovery rate of 1 means LGD = 0, CVA is expected to be 0.

16. Recovery Rate is increased by 30% assuming there is no change to the default rate

    - Assuming there is no change the default rate, CVA is expected to be decreased when the recovery rate is increased. This is because an increase of the recovery rate decreases the LGD, thus CVA is expected to be decreased.

17. Recovery Rate is decreased by 30% assuming there is no change to the default rate

    - Assuming there is no change the default rate, CVA is expected to be increased when the recovery rate is decreased. This is because a decrease of the recovery rate increases the LGD, thus CVA is expected to be increased

18. CDS spreads are increased by 50bps

    - CVA is expected to be increased when CDS spreads are increased. This is because an increase of the CDS spreads increases the PD, thus CVA is expected to be increased.

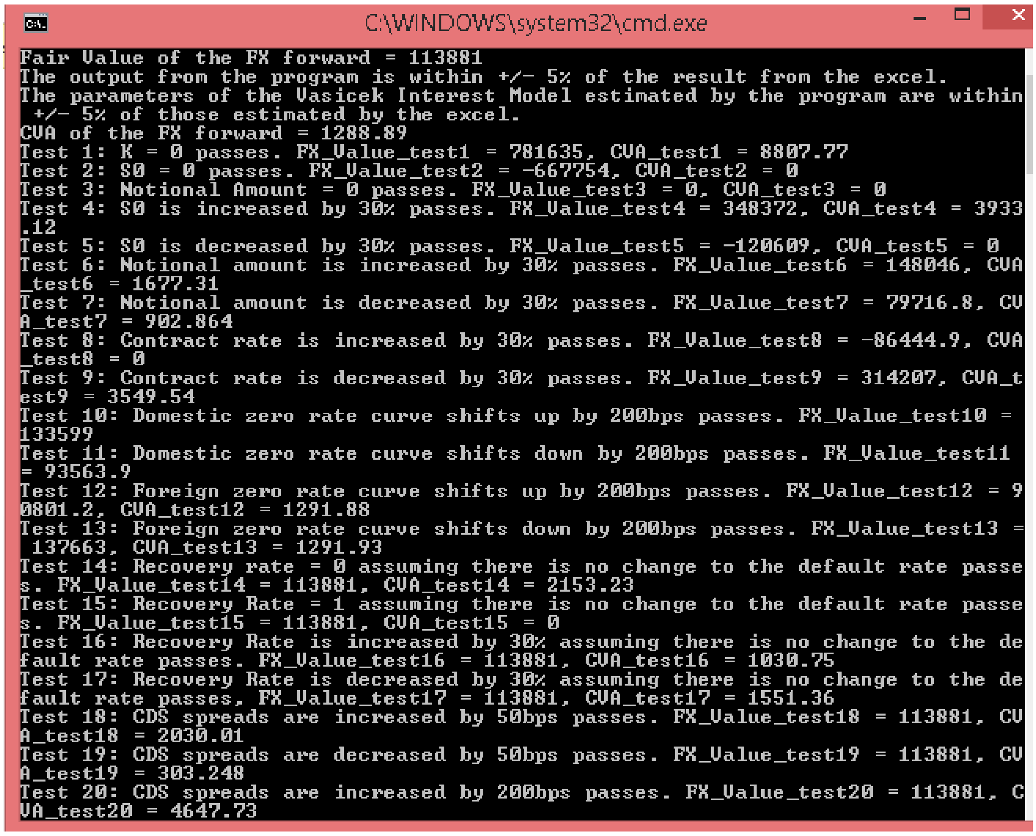
19. CDS spreads are decreased by 50bps

    - CVA is expected to be decreased when CDS spreads are decreased. This is because a decrease of the CDS spreads decreases the PD, thus CVA is expected to be decreased.

20. CDS spreads are increased by 200bps passes

    - CVA is expected to be increased when CDS spreads are increased. This is because an increase of the CDS spreads increases the PD, thus CVA is expected to be increased.

Below is the testing result of the program:



5.4. Validation Summary and Model Restriction

Test of Assumptions	**Observation and Remediation**
FX follows geometric Brownian Motion	The excess skewness and kurtosis are noted.
Interest rate follows Vasicek Model	It is noted that the model cannot capture twisting and jumping of yield curve.
Recovery rate is 40%	The assumption is not considered unreasonable. The effect is not material.
Independence between hazard rate and EAD	Stochastic hazard rate model will be considered.
Implementation Test	**Status**
FX Forward	13 tests passed
Vasicek Model Calibration	Passed
Hazard Rate Model	7 tests passed
CVA Directional Test	20 tests passed

The model is approved for use with the following restriction.

Model Restriction

1. The model cannot be used in stress period.

Model Limitation

1. The model does not perform well when the FX's skewness/ kurtosis is high.

2. The hazard rate is deterministic, which cannot capture the wrong way risk and perform CVA VaR calculation.

3. The model cannot capture the jumping of interest rate.

6. Future Improvement

EAD Model

1. The drift and volatility of the Geometric Brownian Motion are user inputs.

    - Include a calibration of the Geometric Brownian Motion process to estimate the drift term

    - Use the implied volatility from option pricing or volatility estimation from a GARCH model

2. The program can accommodate more products.

Interest rate model

3. Term structure of interest rate can be constructed.

4. Consider the correlation structure of FX and interest rates in the Monte Carlo Simulation. StochasticProcessArray and multipath classes of Quantlib can be used.

Hazard rate model

5. A stochastic intensity model can be constructed so that CVAVaR can be calculated and wrongway risk can be captured.

6. Bilateral CVA can be computed by including a default time calculator.

Interface

7. C++ can accept data input from a csv file. We have used this method to import the data used to bootstrap the zero rates and calibrate the Vasicek interest rate model into C++. We can design a use input interface with Microsoft Excel and the user can input all the information (e.g. notional amount, contract rate, currencies, maturity date, valuation date etc.) into the spreadsheet. Then the C++ will automatically read all the user inputs and pass them into the program.

Performance

8. Apply American Monte Carlo simulation to simulate the CVA.

9. The declaration of variables and flow of the program can be revised. This could enhance the computation efficiency and shorten the computation time.

10. The model testing code uses the CVA calculation code repeatedly. The CVA calculation part can be converted into a function. This can make the model testing codes cleaner and tidier, thus it is more user-friendly.

11. We can employ antithetic variates to increase accuracy by reducing simulation variance. It can also shorten computation time. The enabler is the monotonic pricing function of FX forward. The variance reduction is larger when the spot is higher than the contract price. The antithetic variates must be applied cross section (at every time step) but not along the time steps, otherwise will result in bias. This technique will be particular useful for calculation portfolio CVA.

Appendix – Antithetic Variates

- Standard Monte Carlo Estimator $\theta = {}^{1}/_{2n} \sum_{i=1}^{2n} f(X_i)$
- $Var(\theta) = \sigma^2 / 2n$
- Antithetic Variates Estimator $\theta_{AN} = {}^{1}/_{2n} \sum_{i=1}^{n} \{f(X_i) + f(Y_i)\}$,
- $Var(\theta_{AN}) = \sigma^2 (1+\rho)/2n$, where $\rho = Corr(f(X), f(Y))$
- *If $f(X_1, \ldots, X_n)$ is a monotone function, $\rho \leq 0$*
- Max(MTM of FX Forward,0) is a monotone function and therefore antithetic variates can play a role here.

Below is a case showing the variance reduction of the antithetic variates.

```
> N <- 100000
> S0 <- 80
> K <- 100
> mu <- 0.05
> sigma <- 0.3
> T <- 1
>
> ## Standard Monte Carlo##
> z1 <- rnorm(2*N)
> z2 <- z1[1:N]
> z3 <- -z2
>
> St <- S0*exp((mu-0.5*sigma^2)*T+sigma*sqrt(T)*z1)
> payoff <- pmax(St-K*exp(-(mu-0.5*sigma^2)*T),0)
> mean(payoff)
[1] 4.908415
> var1=var(payoff)/(2*N)
> var1
[1] 0.0008392586
>
> ## Antithetic variates##
>
> St1<- S0*exp((mu-0.5*sigma^2)*T+sigma*sqrt(T)*(z2))
> St2<-S0*exp((mu-0.5*sigma^2)*T+sigma*sqrt(T)*(z3))
> payoff1<- pmax(St1-K*exp(-(mu-0.5*sigma^2)*T),0)
> payoff2<- pmax(St2-K*exp(-(mu-0.5*sigma^2)*T),0)
>
> payoffat <- (payoff1+payoff2)/2
> mean(payoffat)
[1] 4.906083
> var2=var(payoffat)/N
> var2
[1] 0.0007199774
>
> #Variance Reduction (%)#
> (var2/var1-1)*100
[1] -14.21269
```

```
> N <- 100000
> S0 <- 100
> K <- 100
> mu <- 0.05
> sigma <- 0.3
> T <- 1
>
> ## Standard Monte Carlo##
> z1 <- rnorm(2*N)
> z2 <- z1[1:N]
> z3 <- -z2
>
> St <- S0*exp((mu-0.5*sigma^2)*T+sigma*sqrt(T)*z1)
> payoff <- pmax(St-K*exp(-(mu-0.5*sigma^2)*T),0)
> mean(payoff)
[1] 15.22564
> var1=var(payoff)/(2*N)
> var1
[1] 0.002829882
>
> ## Antithetic variates##
>
> St1<- S0*exp((mu-0.5*sigma^2)*T+sigma*sqrt(T)*(z2))
> St2<-S0*exp((mu-0.5*sigma^2)*T+sigma*sqrt(T)*(z3))
> payoff1<- pmax(St1-K*exp(-(mu-0.5*sigma^2)*T),0)
> payoff2<- pmax(St2-K*exp(-(mu-0.5*sigma^2)*T),0)
>
> payoffat <- (payoff1+payoff2)/2
> mean(payoffat)
[1] 15.17599
> var2=var(payoffat)/N
> var2
[1] 0.001675968
>
> #Variance Reduction (%)#
> (var2/var1-1)*100
[1] -40.77606
```

7. User Manual

The program needs two calibration processes to work properly:

- ✓ Domestic and foreign zero rates of various tenors;
- ✓ The parameters for Vasicek interest rate model.

Both calibrations are based on external data. It would be unrealistic to let users input all the data one by one in the interface. The program should be able to read the data from external file provided by the user and do the rest of job. The following sections will elaborate how to prepare these data files.

7.1. Bond information file

The program will read the bond prices and coupon data in order to calibrate the desired zero rates. Users can search for the data on the market and organize them in a CSV file. Below is an example of the CSV file:

Number of observations	Domestic_BondPrice	Domestic_Coupon	Foreign_BondPrice	Foreign_Coupon	Tenor	Tenor_description
1	99.95	0	99.95	0	0.0028	ON
2	99.9	0	99.89	0	0.25	3 mon
3	99.85	0	99.83	0	0.5	6 mon
4	99.8	5	99.77	4	1	1Y
5	99.75	3	99.71	3	1.5	1.5Y
6	99.7	3	99.65	3	2	2Y
7	99.65	5	99.59	3	2.5	2.5Y
8	99.6	3	99.53	3	3	3Y
9	99.55	4	99.47	3	3.5	3.5Y
10	99.5	4	99.41	4	4	4Y
11	99.45	3	99.35	4	4.5	4.5Y
12	99.4	4	99.29	4	5	5Y
13	99.35	4	99.23	4	5.5	5.5Y
14	99.3	5	99.17	3	6	6Y
15	99.25	3	99.11	3	6.5	6.5Y
16	99.2	4	99.05	3	7	7Y
17	99.15	5	98.99	5	7.5	7.5Y
18	99.1	3	98.93	3	8	8Y
19	99.05	5	98.87	3	8.5	8.5Y
20	99	4	98.81	5	9	9Y
21	98.95	5	98.75	4	9.5	9.5Y
22	98.9	4	98.69	4	10	10Y

Column description (Blue fields are required, orange fields are optional) :

Number of observations: This should start from 1.

Domestic_BondPrice: The domestic bond price of corresponding tenor.

Domestic_Coupon: The domestic bond coupon of corresponding tenor. Note that the Bond of tenor less than 1 year does not pay coupon, so the coupon for first three tenors (ON, 3 Mon, 6 Mon) is fixed to be 0.

Foreign_BondPrice: Same as "Domestic_BondPrice", except that the prices are for foreign bonds.

Foreing_Coupon: Same as "Foreign_BondPrice", except that the coupons are for foreign bonds.

Tenor: The tenor is measured in year, starting from ON (Over Night).

Name the CSV file as "BootStrapping.csv" (or other names as long as the name matches the one in program) and put the file under the same directory with the CPP file.

7.2. Interest short rate information file

The program will read the interest short rate information to calibrate the parameters in Vasicek model. In our program we use monthly short rate. User can change it to daily rate as long as the time increment variable “delta_t” in the program is set to match the short rate. Below we give an example of the CSV file.

Number of observations	Int_rate	Int_rate_lag
1	0.24285	0.23985
2	0.23985	0.23935
3	0.23935	0.2421
4	0.2421	0.2404
5	0.2404	0.24165
6	0.24165	0.24165
7	0.24165	0.2389
8	0.2389	0.23675
9	0.23675	0.23785
10	0.23785	0.23635
11	0.23635	0.2366
12	0.2366	0.2371
13	0.2371	0.2366
14	0.2366	0.2371
15	0.2371	0.2386
16	0.2386	0.23535
17	0.23535	0.2361
18	0.2361	0.2361
19	0.2361	0.2356
20	0.2356	0.2376
21	0.2376	0.2366
22	0.2366	0.2356
23	0.2356	0.23645

Column descriptions (All three columns are required):

Number of observations: This should start from 1.

Int_rate: The short rate data from the market.

Int_rate_lag: The short rate data with one-time lag just to facilitate the calibration. Note that as this field is short rate data with one time lag of the last field (i.e. Int_rate), the last record of this field could be set to be empty.

Name this file as “data2.csv” (or other names as long as the name matches the one in program) and put the file under the same directory with the CPP file.

Both “BootStrapping.csv” and “data2.csv” are attached here for reference.

BootStrapping.csv   data2.csv