Cutting CVA's complexity

February 18, 2016

Premia 18

The following method computes the credit value adjustment (CVA in the following) by using marked branching diffusions. It is based on the paper [1].

1 Introduction

This method computes the price of a Credit Value Adjustment in a Markovian model, in the presence of counterparty-risk or not. In each case, the mark-to-market value satisfies a non-linear PDE. The method developed to solve these PDEs is based on branching diffusions describing a marked Galton-Watson random tree.

2 Theoretical framework

We assume the issuer is allowed to dynamically trade one dimensional underlying asset X. Additionally, to hedge his credit risk on the counterparty name, he can trade a default risky bond, denoted P_t. The values of the underlyings are not altered by the counterparty default which is modeled by a Poisson jump process with constant intensity. We consider a long position in a single derivative denoted u, whose payoff is denoted Ψ. The processes X and P satisfy under the risk-neutral measure $\mathbb{P}$

$$\begin{aligned} dX_t &= r\,X_t dt + \sigma\,X_t dW_t, X_0 = x \\ dP_t &= (r+\lambda)\,P_t dt - P_t dJ_t, \end{aligned}$$

where W is a Brownian motion and J is a jump Poisson process with intensity λ and r is the interest rate. The no arbitrage condition and the completeness of the market give that $e^{-rt}u(t,X_t)$ is a $\mathbb{P}$-martingale characterized by

$$\partial_t u + \mathcal{L}u + \lambda(\tilde{u} - u) - ru = 0,$$

where $\mathcal{L}$ is the Itô generator of X and $\tilde{u}$ is the derivative value after the counterparty has defaulted. At the default event, $\tilde{u} = RM^+ - M^-$ where M is the mark-to-market value of the derivative to be used in the unwinding of the position upon default and R is the recovery rate.

- When the mark-to-market value is calculated with provision for counterparty risk, $M = u$:

$$\partial_t u + \mathcal{L}u - (1-R)\lambda u^+ - ru = 0,\ u(T,x) = \Psi(x)$$

- When the mark-to-market value is calculated without provision for counterparty risk:

$$\partial_t u + \mathcal{L}u + \lambda(RM^+ - M^- - u) - ru = 0,\ u(T,x) = \Psi(x)$$
$$\partial_t M + \mathcal{L}M - rM = 0,\ M(T,x) = \Psi(x).$$

By discounting and replacing u by $-u$, the two PDEs can be rewritten :

$$\partial_t u + \mathcal{L}u + \beta(u^+ - u) = 0,\ u(T,x) = \Psi(x) \text{ PDE2}$$
$$\partial_t u + \mathcal{L}u + \frac{\beta}{1-R}\Big((1-R)(\mathbb{E}_{t,x}(\Psi))^+ + R\mathbb{E}_{t,x}(\Psi) - u\Big) = 0, u(T,x) = \Psi(x) \text{ PDE1}$$

with $\beta = \lambda(1-R)$.

3 Market Branching diffusion

3.1 Case PDE2

In order to derive an algorithm solving PDE2, we first introduce the following PDE

$$\partial_t u + \mathcal{L}u + \beta(F(u) - u) = 0, u(T,x) = \Psi(x), \tag{1}$$

where $F(u) = \sum_{k=0}^M a_k u^k = \sum_{k=0}^M \frac{a_k}{p_k} p_k u^k$, where $\sum_{k=0}^M p_k = 1$ and $\forall k,\ p_k \in [0,1]$.

In order to approximate PDE2 by an equation of type (1), we have to approximate u^+ by a polynomial decomposition. Referring to [1, Equation (21)], we approximate u^+ by

$$F(u) := 0.0589 + 0.5u + 0.8164u^2 - 0.4043u^4. \tag{2}$$

Galton-Watson tree The probabilistic interpretation of equation (1) is the following. Let a single particle start at the origin, perform an Itô diffusion with generator $\mathcal{L}$, and after an exponential time with mean $\frac{1}{\beta}$ (independent of X) die and produce k descendants with probability p_k. Then, independently each descendant performs a Itô diffusion with generator $\mathcal{L}$, die and produce k descendants with probability p_k, and so on until T (If there are N_t particles at time t, the first time a particle dies follows an exponential law of parameter $N_t\beta$). We denote by $Z_t = (z_t^1, \cdots, z_t^{N_t})$ the positions of the N_t particles alive at time t. From [1], we get that

$$\hat{u}(t,x) := \mathbb{E}_{t,x}\left[\prod_{k=0}^M \left(\frac{a_k}{p_k}\right)^{\omega_k} \prod_{i=1}^{N_T} \Psi(z_T^i)\right],$$

is the unique viscosity solution of(1). ω_k denotes, for each Galton-Watson tree, the number of particles that branch into k descendants with $k \in \{0, \cdots, M\}$. $1 + \sum_{k=0}^M (k-1)\omega_k$ gives the total number of individuals produced by the branching $\omega = (\omega_0, \cdots, \omega_M)$. These descendants are drawn with distribution p_k given by

$$p_k := \frac{|a_k| \|\Psi\|_\infty^k}{\sum_{i=0}^M |a_i| \|\Psi\|_\infty^i}.$$

This choice of probability is optimal for convergence (see [1, page 70]).

3.2 Case PDE1

We assume that the function $(1-R)x^+ + Rx$ can be well approximated by a polynomial function $F(x)$. We consider the following PDE, which approximates PDE1

$$\partial_t u(t,x) + \mathcal{L}u(t,x) + \frac{\beta}{1-R}\left(F\left(\mathbb{E}_{t,x}[\Psi(X_T)]\right) - u(t,x)\right) = 0, u(T,x) = \Psi(x).$$

From Feynman Kac formula, we have

$$u(t,x) = \mathbb{E}_{t,x}[\mathbb{1}_{\{\tau \geq T\}}\Psi(X_T)] + \mathbb{E}_{t,x}[\mathbb{1}_{\{\tau<T\}}F(\mathbb{E}_\tau(\Psi(X_T)))]$$

where τ is a Poisson default time with intensity $\frac{\beta}{1-R}$. The term $\mathbb{E}_{t,x}[\mathbb{1}_{\{\tau<T\}}F(\mathbb{E}_\tau(\Psi(X_T)))]$ can be calculated using the previous algorithm by imposing that the particle can default only once.

4 Algorithm

The algorithm for solving PDE1 and PDE2 is the following

1. Choose a polynomial approximation of $u^+ \sim \sum_{k=0}^M a_k u^k$ on the domain $[-1,1]$
2. For each Monte Carlo simulation
   (a) simulate the asset and the first Poisson default time with intensity β (resp. $\frac{\beta}{1-R}$) for PDE2 (resp. for PDE1)
   (b) at each default time, produce k descendants with probability p_k. For PDE1, descendants produced after the first default become immortal
   (c) evaluate the quantities

$$\prod_{k=0}^M \left(\frac{a_k}{p_k}\right)^{\omega_k} \prod_{i=1}^{N_T} \Psi(Z_T^i) \text{ for PDE2}$$

$$\left(\frac{a_1(1-R)+R}{p_1}\right)^{\omega_1} \prod_{k\neq 1}^M \left(\frac{a_k(1-R)}{p_k}\right)^{\omega_k} \prod_{i=1}^{N_T \in [0,M]} \Psi(Z_T^i) \text{ for PDE1}$$

In case of PDE1, $\sum_{k=0}^M \omega_k = 0$ or 1.

5 Implementation and numerical experiments

We test the algorithm with the following parameters :

r	x	σ	R	$\Psi(x)$	K	β	MC
0.05	0.7	0.2	0.4	$1 - 2\mathbb{1}_{\{x>K\}}$	1	0.03	2^{22}

MC represents the number of Monte Carlo simulations.

We compute the value of the mark to market value at time $t = 0$, for different values of T

T	PDE1	PDE2
2	0.7040	0.7031
10	0.1060	0.1131

References

[1] P. Henry-Labordère. Cutting CVA's complexity *Risk Magazine*, p 67-73, 2012. 1, 2