

Pricing synthetic CDO - Premia 9

Vincent Lemaire

Premia 14

Collateralized Debt Obligations (CDO) are credit derivatives *i.e.* financial contracts allowing the transfer of credit risk from one market participant to another. This feature improves the efficient allocation of credit risk among market participants.

1 CDS and CDO

A Credit Default Swap is a contract in which party A pays B a regular cash flow till maturity in exchange for a compensation payment from party B in an event of default of the underlying corporate bond. The cash flow as a percentage of the notional, aka *credit spread*, is determined in such a way that the contract is worth 0 at initiation of the contract. The cash flow reflects the probability of the event of default that two parties agreed upon.

A synthetic CDO is a pool of CDS, of which the cumulative loss on the pool is divided into different *tranches*. A tranche holder receives regular cash flow as a percentage of the remaining balance of that tranche and pays out as loss occurs for which that tranche is responsible till maturity. For example, the holder of the 3% – 7% tranche gets quarterly cashflow as a percentage of the balance. When the total loss of the pool exceeds 3%, the balance of 3% – 7% starts to reduce. When the total loss reaches 7%, the balance of that tranche is gone. In this pricing module, the percentage is absolute and not relative to the total loss.

The cash flows of all the tranches are determined as the same way as CDS and they reflect the joint distribution of the default events of all the underlying CDS contracts.

2 Modelling and factor approach

We will thereafter consider a synthetic CDO with some given maturity T . This is based upon n CDS with nominals $N_j, j = 1, \dots, n$ and maturity also equal to T . We denote by δ_j the *recovery rate* for credit j and by $M_j = (1 - \delta_j)N_j$ the corresponding *loss given default*.

For the n names in the collateral pool, we consider the associated default times τ_1, \dots, τ_n defined on a common probability space $(\Omega, \mathcal{G}, \mathbb{P})$. In the following, we will consider only reduced-form models of default times defined by

$$\tau_i = \inf \left\{ u \in \mathbb{R}^+, \int_0^u h_i(v) dv \geq -\log(U_i) \right\}, \quad (H_\tau)$$

where the h_i are deterministic and continuous positive functions, the U_i are some uniform random variables.

In order to compute (by a semi-analytic approach) the price of one CDO tranche, all we need is the portfolio loss distribution *i.e.* the portfolio aggregate loss on the credit portfolio at time t :

$$L(t) = \sum_{j=1}^n M_j \mathbf{1}_{\{\tau_j \leq t\}}$$

which is a pure jump process. This distribution depend on the joint distribution of the default times τ_1, \dots, τ_n that we modelling using a classical factor approach and Copula functions.

We denote by F and S respectively the *joint distribution* and survival functions such that for all $(t_1, \dots, t_n) \in [0, T]^n$, $F(t_1, \dots, t_n) = \mathbb{P}(\tau_1 \leq t_1, \dots, \tau_n \leq t_n)$ and $S(t_1, \dots, t_n) = \mathbb{P}(\tau_1 > t_1, \dots, \tau_n > t_n)$. F_1, \dots, F_n represent the *marginal* distribution functions and S_1, \dots, S_n the corresponding survival functions. By the assumption (H_τ), we have

$$S_i(t) = \mathbb{P}(\tau_i > t) = \exp\left(-\int_0^t h_i(v) dv\right).$$

We will consider now a latent factor V such that conditionally on V , the default times are independent. We will denote by $p_t^{i|V} = \mathbb{P}(\tau_i \leq t|V)$ and $q_t^{i|V} = 1 - p_t^{i|V}(t)$ the conditional default and survival probabilities. It is easy to check that

$$S(t_1, \dots, t_n) = \int \prod_{i=1}^n q_t^{i|v} f(v) dv,$$

$$F(t_1, \dots, t_n) = \int \prod_{i=1}^n p_t^{i|v} f(v) dv.$$

So, if we can easily compute the conditional default probabilities and integrate along the density of the factor V , we are able to compute the joint distribution of the default times.

2.1 Gaussian Copula

We consider a standard Gaussian random variable V , and we define the Gaussian vector (X_1, \dots, X_n) by

$$X_i = \rho V + \sqrt{1 - \rho^2} \bar{V}_i$$

where \bar{V}_i are *independent* ($\forall i, j, V_i \perp V_j$ and $\forall i, V_i \perp V$) standard Gaussian random variables. We define the uniform random variable $U_i = 1 - \Phi(X_i)$ where Φ is the cumulative distribution function of a standard Gaussian variable. We then get

$$p_t^{i|V} = \Phi\left(\frac{\Phi^{-1}(F_i(t)) - \rho V}{\sqrt{1 - \rho^2}}\right).$$

with $F_i(t) = 1 - \exp(-\int_0^t h_i(v) dv)$.

With the CDO pricing library for Premia, we initialize a one-factor Gaussian copula with parameter $\rho = 0.4$ on the following way

```
copula      *cop;
double      rho = 0.4;
cop = init_gaussian_copula(rho);
```

2.2 Clayton Copula

We consider a positive random variable V following a standard Gamma distribution with shape parameter $1/\theta$ where $\theta > 0$. Its probability density is given by $f(x) = \frac{1}{\Gamma(1)} \exp(-x)x^{(1-\theta)*\theta}$ for $x > 0$. We define the uniform random variable

$$U_i = \Psi\left(-\frac{\log(\bar{U}_i)}{V}\right),$$

where $\bar{U}_1, \dots, \bar{U}_n$ are independent uniform random variables also independent from V , and Ψ is the Laplace transform of f . The joint distribution of (U_1, \dots, U_n) is known as the Clayton copula.

The conditional default probabilities can be expressed as

$$p_t^{i|V} = \exp\left(V(1 - F_i(t))^{-\theta}\right),$$

with $F_i(t) = 1 - \exp(-\int_0^t h_i(v) dv)$.

In Premia, we initialize a one-factor Clayton copula with parameter $\rho = 0.4$ like this:

```
copula      *cop;
cop = init_clayton_copula(0.40);
```

2.3 NIG Copula

The normal inverse Gaussian distribution (NIG) is a mixture of normal and inverse Gaussian distributions. The density of a random variable $X \sim \mathcal{NIG}(\alpha, \beta, \mu, \delta)$ is given by:

$$f_{\mathcal{NIG}}(x; \alpha, \beta, \mu, \delta) = \frac{\delta \alpha \exp(\delta \gamma + \beta(x - \mu))}{\pi \sqrt{\delta^2 + (x - \mu)^2}} K_1(\alpha \sqrt{\delta^2 + (x - \mu)^2}),$$

where $K_1(w) = \frac{1}{2} \int_0^\infty \exp(-\frac{1}{2}w(t+t^{-1}))dt$ is the modified Bessel function of the third kind.

We consider a random variable V following a NIG distribution with parameters

$$V \sim \mathcal{NIG}(\alpha, \beta, -\frac{\beta\gamma^2}{\alpha^2}, \frac{\gamma^3}{\alpha^2})$$

where $\gamma = \sqrt{\alpha^2 - \beta^2}$, and we define

$$X_i = \rho V + \sqrt{1 - \rho^2} \bar{V}_i,$$

where \bar{V}_i are independent (and independent from V) NIG random variables with parameters

$$\bar{V}_i \sim \mathcal{NIG}(\frac{\sqrt{1-\rho^2}}{\rho}\alpha, \frac{\sqrt{1-\rho^2}}{\rho}\beta, -\frac{\sqrt{1-\rho^2}}{\rho}\frac{\beta\gamma^2}{\alpha^2}, \frac{\sqrt{1-\rho^2}}{\rho}\frac{\gamma^3}{\alpha^2}).$$

To simplify notations we denote $F_{\mathcal{NIG}(s)}(x)$ the cumulative distribution of a NIG random variable with parameters $\mathcal{NIG}(s\alpha, s\beta, -s\frac{\beta\gamma^2}{\alpha^2}, s\frac{\gamma^3}{\alpha^2})$. Using the scaling property and stability under convolution of NIG distribution we get $X_i \sim \mathcal{NIG}(1/\rho)$. We define the uniform random variable $U_i = 1 - F_{\mathcal{NIG}(1/\rho)}(X_i)$ and we get

$$p_t^{i|V} = F_{\mathcal{NIG}(\sqrt{1-\rho^2}/\rho)}\left(\frac{F_{\mathcal{NIG}(1/\rho)}^{-1}(F_i(t)) - \rho V}{\sqrt{1-\rho^2}}\right).$$

The initialization of the NIG copula is done by the following lines of code:

```
copula      *cop;
double      rho = 0.4, alpha = 1.2, beta = -0.2;
cop = init_nig_copula(rho, alpha, beta);
```

3 Number of defaults

We thereafter denote by

$$N(t) = \sum_{i=1}^n \mathbf{1}_{\{\tau_i \leq t\}},$$

the counting process associated with the number of defaults. In case where the CDO is homogeneous (*i.e.* $\forall i \in \{1, \dots, n\}, N_i = N$, δ deterministic and $\delta_i = \delta$), the portfolio aggregate loss on the credit portfolio at time t is given by

$$L(t) = N(1 - \delta)N(t).$$

3.1 Hull and White recursion

The counting process $(N(t))_{t \in [0, T]}$ is a discrete random process with values in $\{0, \dots, n\}$. For a fixed time t , we denote by $\pi_k^{i|V} = \mathbb{P}(N(t) = k|V)$ and calculate $\pi_k^{i|V}$ iteratively. First, the conditional probability of no-default is

$$\pi_0^{i|V} = \mathbb{P}\left(\sum_{i=1}^n \mathbf{1}_{\tau_i \leq t} = 0|V\right) = \prod_{i=1}^n \mathbb{P}(\tau_i > t|V) = \prod_{i=1}^n q_t^{i|V}.$$

The probability of one default at time t is

$$\begin{aligned} \pi_1^{i|V} &= \mathbb{P}\left(\bigcup_{j=1}^n \left\{\tau_j \leq t\right\} \cap \left(\bigcap_{i=1, i \neq j}^n \left\{\tau_i > t\right\}\right)\right), \\ &= \sum_{i=1}^n \left((1 - q_t^{i|V}) \prod_{j=1, j \neq i}^n q_t^{j|V}\right). \end{aligned}$$

Iteratively, we prove that the conditional probability of exactly k defaults by time t is

$$\pi_k^{i|V} = \pi_0^{i|V} \sum w_{z(1)} \dots w_{z(k)}$$

where $w_i = \frac{p_t^{i|V}}{q_t^{i|V}}$ and $\{z(1), \dots, z(k)\}$ is a set of k different numbers chosen from $\{1, \dots, N\}$. A relatively fast way to computing this summation (taken over C_k^n different ways) is to use the following recurrence relation between $U_k = \sum w_{z(1)} \dots w_{z(k)}$ and $V_k = \sum_{j=1}^n w_j^k$:

$$\begin{aligned} U_1 &= V_1, \\ 2U_2 &= V_1U_1 - V_2, \\ 3U_3 &= V_1U_2 - V_2U_1 + V_3, \\ &\vdots \\ kU_k &= V_1U_{k-1} + V_2U_{k-2} - \dots + (-1)^k V_{k-1}U_1 + (-1)^{k+1} V_k. \end{aligned}$$

This recursion is implemented in the function `hw_numdef` in file `hull_white.c`. In the following example, we initialize a time grid `t`, compute the conditional probabilities $p_t^{i|V}$ in `cp` and compute the number of defaults distribution by the recursive algorithm.

```
t = init_fine_grid(cdo->dates, 10);
cp = init_cond_prob(cdo, cop, t);
numdef = hw_numdef(cdo, cop, t, cp);
```

3.2 Laurent and Gregory approach

In this approach, we compute the probability generating function of $N(t)$

$$\psi_{N(t)}(u) = \mathbb{E}(u^{N(t)}) = \sum_{k=0}^n \mathbb{P}(N(t) = k) u^k.$$

By the conditional Independence of the default time we get

$$\psi_{N(t)}(u) = \int \underbrace{\prod_{i=1}^n (q_t^{i|v} + p_t^{i|v} u)}_{u^n \phi_n(v) + \dots + \phi_0(v)} f(v) dv.$$

The formal expansion of $\prod_{i=1}^n (q_t^{i|v} + p_t^{i|v} u)$ is very effective for $n < 200$. We can then obtain the probability of k names being in default at time t as

$$\mathbb{P}(N(t) = k) = \int \phi_k(v) f(v) dv.$$

The computation is done by the function `lg_numdef` in file `laurent_gregory.c`

```
t = init_fine_grid(cdo->dates, 10);
cp = init_cond_prob(cdo, cop, t);
numdef = lg_numdef(cdo, cop, t, cp);
```

4 Loss distribution and cumulative loss

We recall that the loss distribution at time t is defined by

$$L(t) = \sum_{j=1}^n M_j \mathbf{1}_{\{\tau_j \leq t\}} = \sum_{j=1}^n N_j (1 - \delta_j) \mathbf{1}_{\{\tau_j \leq t\}}$$

where N_j is deterministic and δ_j is the recovery rate (possibly random) taking value in $[0, 1]$.

Denote therefore by $M_{A,B}(t)$ the cumulative loss on a given tranche $[A, B]$ at time t :

$$M_{A,B}(t) = \begin{cases} 0, & \text{if } L(t) \leq A, \\ L(t) - A, & \text{if } L(t) \in [A, B], \\ B - A, & \text{if } L(t) \geq B. \end{cases}$$

The expected cumulative loss on a given tranche at time t can be obtained from the loss distribution on the portfolio of reference credits

$$\begin{aligned}\mathbb{E}(M_{A,B}(t)) &= (B - A)\mathbb{P}_{L(t)}(\textstyle\int B, +\infty] + \int_A^B (x - A)\mathbb{P}_{L(t)}(dx), \\ &= \mathbb{E}((L(t) - A)_+ - (L(t) - B)_+),\end{aligned}$$

where $\mathbb{P}_{L(t)}$ is the distribution of $L(t)$.

Remark 1 (Homogeneous CDO). *For an homogeneous CDO we deduce immediately the loss distribution $L(t)$ from the number of defaults distribution $N(t)$. Moreover, an approximation of $\mathbb{E}(M_{A,B}(t))$ is*

$$\mathbb{E}(M_{A,B}(t)) \simeq \sum_{k=0}^n \left((N(1-\delta)k - A)_+ - (N(1-\delta)k - B)_+ \right) \mathbb{P}(N(t) = k),$$

This approximation is done in function `mean_losses_from_numdef` in file `cdo.c`.

```
meanloss = mean_losses_from_numdef(cdo, t, numdef);
```

4.1 Recursive algorithm

Recursive approach for the Homogeneous case

If the CDO is homogeneous, the loss $(L(t))_{t \in [0, T]}$ is a discrete random process with values in $\{0, \frac{1}{N}, \dots, 1\}$. For a fixed time t , we denote by $\pi_k^{(n)|V} = \mathbb{P}(L(t) = k/(N(1-\delta))|V)$ and calculate $\pi_k^{(n)|V}$ iteratively by first assuming that there are no debt instruments ($\pi_k^{(0)|V}$), then assuming that there is only one debt instrument ($\pi_k^{(1)|V}$), then assuming that there are only two debt instruments and so on.

When there are no debt instruments we are certain there will be no loss. Hence

$$\pi_0^{(0)|V} = 1 \quad \text{and} \quad \pi_k^{(0)|V} = 0, \forall k \geq 1.$$

Suppose that we have calculated $\pi_k^{(j)|V}$ when the first j debt instruments are considered. Then for every $k \in \{0, \dots, n\}$

$$\begin{aligned}\pi_k^{(j+1)|V} &= \mathbb{P}\left(L^{(j+1)}(t) = \frac{k}{N(1-\delta)}|V\right), \\ &= \mathbb{P}\left(L^{(j)}(t) = \frac{k-1}{N(1-\delta)}|V\right) \mathbb{P}(\tau_{j+1} \leq t|V) + \mathbb{P}\left(L^{(j)}(t) = \frac{k}{N(1-\delta)}|V\right) \mathbb{P}(\tau_{j+1} > t|V) \\ &= \pi_{k-1}^{(j)|V} p_t^{j+1|V} + \pi_k^{(j)|V} q_t^{j+1|V}.\end{aligned}$$

When all n debt instruments have been considered we obtain the conditional loss distribution at time t . The loss distribution is then given by

$$\mathbb{P}(L(t) = k) = \int \pi_k^{(n)|v} f(v) dv.$$

This algorithm is implemented in the function `hw_losses_h` in the file `hull_white.c`. We can call this function by the following way:

```
t = init_fine_grid(cdo->dates, 4);
x = init_hom_grid(MINDOUBLE, 1-delta, 1/100.); // if n = 100
cp = init_cond_prob(cdo, cop, t);
losses = hw_losses_h(cdo, cop, t, x, cp);
```

Recursive approach for the general case

We assume that the recovery rate is deterministic and the same δ for each debt instrument. It is not necessary for the nominal to be equal. The loss process is a pure jump process taking value in a discrete subset of $[0, 1]$. The key idea is to discretize $[0, 1]$ choosing K buckets $[x_k, x_{k+1}[$ such that $x_0 = 0$ and $x_K = +\infty$.

For a fixed time t , we denote by $\pi_k^{(n)|V} = \mathbb{P}(L(t) \in [x_k, x_{k+1}[|V)$ and $A_k^{(n)|V} = \mathbb{E}(L(t)|L(t) \in [x_k, x_{k+1}[|V)$. As above, we compute $\pi_k^{(n)|V}$ and $A_k^{(n)|V}$ iteratively. Suppose that we have calculated $\pi_k^{(j)|V}$ when the first j debt instruments are considered. Then for every $k \in \{0, \dots, K\}$, we define $u(k)$ as the bucket containing $A_k^{(j)|V} + N_{j+1}(1 - \delta_{j+1})$.

- if $u(k) > k$ then

$$\begin{aligned}\pi_k^{(j+1)|V} &= \pi_k^{(j)|V} - \pi_k^{(j)|V} p_t^{j+1|V}, \\ \pi_{u(k)}^{(j+1)|V} &= \pi_{u(k)}^{(j)|V} + \pi_k^{(j)|V} p_t^{j+1|V}, \\ A_k^{(j+1)|V} &= A_k^{(j)|V}, \\ A_{u(k)}^{(j+1)|V} &= A_{u(k)}^{(j)|V} + \frac{\pi_k^{(j)|V} p_t^{j+1|V}}{\pi_{u(k)}^{(j)|V} + \pi_k^{(j)|V} p_t^{j+1|V}} (A_k^{(j)|V} + N_{j+1}(1 - \delta_{j+1}) - A_{u(k)}^{(j+1)|V}),\end{aligned}$$

- if $u(k) = k$ then

$$\begin{aligned}\pi_k^{(j+1)|V} &= \pi_k^{(j)|V}, \\ A_k^{(j+1)|V} &= A_k^{(j)|V} + p_t^{j+1|V} N_{j+1}(1 - \delta_{j+1}).\end{aligned}$$

This algorithm is implemented in the function `hw_losses_nh` in the file `hull_white.c`.

```
t = init_fine_grid(cdo->dates, 4);
x = init_hom_grid(MINDOUBLE, 1, 1/1000.); // the buckets
cp = init_cond_prob(cdo, cop, t);
losses = hw_losses_nh(cdo, cop, t, x, cp);
```

4.2 FFT approach

In this section, we assume that the σ -algebras $\sigma(\delta_1), \dots, \sigma(\delta_n), \sigma(V, \tau_1, \dots, \tau_n)$ are independent. We compute the characteristic function of the cumulative losses for different time horizons,

$$\phi_{L(t)}(u) = \mathbb{E} \left(\exp \left(iu \sum_{j=1}^n N_j (1 - \tau_j) \mathbf{1}_{\{\tau_j \leq t\}} \right) \right).$$

From the independence of τ_j conditionally on V and the independence assumption over the recovery rates δ_j , we get

$$\phi_{L(t)}(u) = \int \prod_{j=1}^n \left(q_t^{j|V} + p_t^{j|V} \mathbb{E}(\exp(iu N_j (1 - \delta_j))) \right) f(v) dv.$$

We get the distribution of $L(t)$ by the inverse fast Fourier transform (provided by the library `FFTW`¹). This is done by the function `lg_losses` in the file `laurent_gregory.c` who is call, for example, by

```
t = init_fine_grid(cdo->dates, 4);
x = init_hom_grid(MINDOUBLE, 1, 1/1000.); // 1000 points for the FFT
cp = init_cond_prob(cdo, cop, t);
losses = lg_losses(cdo, cop, t, x, cp);
```

5 Price of a CDO tranche

Let

$$0 = t_0 < t_1 < \dots < t_m = T$$

denote the payment dates and $D(t)$ the discount factor for maturity t . We assume that $D(t) = \exp(-\int_0^t r(s) ds)$ where r is a step function or a continuous linear by step function. In the code, `rates` denotes the `step_fun` (see file `structs.h`) function $t \mapsto \int_0^t r(s) ds$.

¹<http://www.fftw.org/>

5.1 Default leg

The value of the default leg (DL) can be calculated as the expected value of the discounted default payments:

$$\begin{aligned} DL &= \mathbb{E} \left(\sum_{j=1}^n \mathbf{1}_{\tau_j \leq T} D(\tau_j) (M_{A,B}(\tau_j) - M_{A,B}(\tau_j^-)) \right), \\ &= \mathbb{E} \left(\int_0^T D(t) dM_{A,B}(t) \right). \end{aligned}$$

We consider a time grid $0 = \tilde{t}_0 < \dots < \tilde{t}_{\tilde{m}} = T$ such that $\forall i \in \{1, \dots, m\}, \exists j \in \{1, \dots, \tilde{m}\}, \tilde{t}_j = t_i$. This time grid is the `grid t` in the code. We approximate the default leg by

$$DL \simeq \sum_{j=0}^{\tilde{m}-1} D\left(\frac{\tilde{t}_j + \tilde{t}_{j+1}}{2}\right) (\mathbb{E}(M_{A,B}(\tilde{t}_{j+1})) - \mathbb{E}(M_{A,B}(\tilde{t}_j))).$$

This is done by the function

```
dl = default_leg(cdo, rates, t, meanloss);
```

in file `cdo.c`. We recall that the grid `meanloss` contains $\{\mathbb{E}(M_{A,B}(\tilde{t}_j))\}_{j=0, \dots, \tilde{m}}$.

5.2 Payment leg

The value of the payment leg (or premium leg) (PL) of the tranche is

$$PL = \underbrace{\mathbb{E} \left(\sum_{j=1}^n \mathbf{1}_{\tau_j \leq T} X D(\tau_j) (M_{A,B}(\tau_j) - M_{A,B}(\tau_j^-)(\tau_j - t_{k(j)})) \right)}_{\text{accrued margin}} + \underbrace{\sum_{i=1}^m X D(t_i) (B - A - \mathbb{E}(M_{A,B}(t_i)))}_{\text{payments made at regular payment dates}},$$

where X is the premium of the tranche $[A, B]$. The valuation of the accrued margins is computed as above, using the same time grid $0 = \tilde{t}_0 < \dots < \tilde{t}_{\tilde{m}} = T$ and we obtain the following approximation of the payment leg

$$PL \simeq \sum_{j=0}^{\tilde{m}-1} D\left(\frac{\tilde{t}_j + \tilde{t}_{j+1}}{2}\right) (\mathbb{E}(M_{A,B}(\tilde{t}_{j+1})) - \mathbb{E}(M_{A,B}(\tilde{t}_j))) (\tilde{t}_{j+1} - \tilde{t}_j) + \sum_{i=1}^m X D(t_i) (B - A - \mathbb{E}(M_{A,B}(t_i))).$$

This is done by the function

```
dl = default_leg(cdo, rates, t, meanloss);
```

in file `cdo.c`.

5.3 Price

The price of the tranche $[A, B]$ is the premium X such that $DL = PL$ *i.e.*

$$X = \frac{\sum_{j=0}^{\tilde{m}-1} D\left(\frac{\tilde{t}_j + \tilde{t}_{j+1}}{2}\right) (\mathbb{E}(M_{A,B}(\tilde{t}_{j+1})) - \mathbb{E}(M_{A,B}(\tilde{t}_j)))}{\sum_{j=0}^{\tilde{m}-1} D\left(\frac{\tilde{t}_j + \tilde{t}_{j+1}}{2}\right) (\mathbb{E}(M_{A,B}(\tilde{t}_{j+1})) - \mathbb{E}(M_{A,B}(\tilde{t}_j))) (\tilde{t}_{j+1} - \tilde{t}_j) + \sum_{i=1}^m X D(t_i) (B - A - \mathbb{E}(M_{A,B}(t_i)))}.$$

6 Monte-Carlo methods

We recall that $DL = \mathbb{E}(Y_{DL})$ where $Y_{DL} = \sum_{j=1}^n \mathbf{1}_{\tau_j \leq T} D(\tau_j) (M_{A,B}(\tau_j) - M_{A,B}(\tau_j^-))$. An approximation of the default leg DL is

$$DL \simeq \frac{1}{K} \sum_{k=1}^K Y_{DL}^{(k)}$$

with

$$Y_{DL}^{(k)} = \sum_{j=1}^n \mathbf{1}_{\{\tau_j^{(k)} \leq T\}} D(\tau_j^{(k)}) (M_{A,B}(\tau_j^{(k)}) - M_{A,B}(\tau_j^{(k)-})).$$

In the same way, an approximation of the payment leg PL is

$$PL \simeq \frac{1}{K} \sum_{k=1}^K XY_{PL}^{(k)}$$

with

$$Y_{PL}^{(k)} = \sum_{j=1}^n \mathbf{1}_{\{\tau_j^{(k)} \leq T\}} D(\tau_j^{(k)}) (M_{A,B}(\tau_j^{(k)}) - M_{A,B}(\tau_j^{(k)-}) (\tau_j^{(k)} - t_{k(j)})) + \sum_{i=1}^m D(t_i) (B - A - M_{A,B}(t_i)).$$

This algorithm is implemented in the functions `mc_payment_leg` and `mc_default_leg`:

```
n_mc = 100000;
pl = mc_payment_leg(cdo, cop, rates, n_mc);
dl = mc_default_leg(cdo, cop, rates, n_mc);
```

6.1 Monte-Carlo with control variate

We consider as control variate an homogeneous CDO built with the following nominal and recovery rate:

$$N^h = \frac{1}{n} \sum_{j=1}^n N_j, \quad \text{and} \quad \delta^h = \frac{1}{n} \sum_{j=1}^n \mathbb{E}(\delta_j).$$

Then we compute the legs of this homogeneous CDO using the Laurent and Gregory approach (see section 3.2). Hence we have $N^h(t)$ the number of default distribution, $L^h(t)$ the loss distribution, and PL^h and DL^h the price of the payment leg and the default leg respectively.

The value of the default leg of the initial CDO is then equal to

$$DL = DL^h + \mathbb{E} \left(\sum_{j=1}^n \mathbf{1}_{\tau_j \leq T} D(\tau_j) (M_{A,B}(\tau_j) - M_{A,B}(\tau_j^-) - (M_{A,B}^h(\tau_j) - M_{A,B}^h(\tau_j^-))) \right),$$

where $M_{A,B}^h(t) = (L^h(t) - A) \mathbf{1}_{L^h(t) \in [A,B]} + (B - A) \mathbf{1}_{L^h(t) \geq B}$. We approximate the expectation by a Monte-Carlo procedure as above. In the same way, we can express the payment leg with PL^h .

This algorithm is implemented in the functions `mc_payment_vc_leg` and `mc_default_vc_leg`:

```
n_mc = 100000;
t = init_fine_grid(cdo->dates, 4);
hcdo = homogenize_CDO(cdo);
cp = init_cond_prob(hcdo, cop, t);
numdef = lg_numdef(hcdo, cop, t, cp);
meanloss = mean_losses_from_numdef(hcdo, t, numdef);
hpl = payment_leg(hcdo, rates, t, meanloss);
hdl = default_leg(hcdo, rates, t, meanloss);
pl = mc_payment_vc_leg(cdo, cop, rates, n_mc);
dl = mc_default_vc_leg(cdo, cop, rates, n_mc);
for (jtr = 0; jtr < cdo->n_tranches-1; jtr++) {
    pl[jtr] = hpl[jtr] + pl[jtr];
    dl[jtr] = hdl[jtr] + dl[jtr];
}
```

References

- [1] L.Andersen, J.Sidenious (2004) *Extension to the Gaussian Copula: Random Recovery and Random Factor Loadings* Preprint
- [2] X.Burtschell, J.P.Laurent, J.Gregory (2005) *A comparative analysis of CDO pricing models*.Preprint.
- [3] J.Hull, A.White (2004) *Valuation of a CDO and an n^{th} to default CDS without Monte Carlo simulation*. Journal of Derivatives,2, 8-23.

- [4] J.P.Laurent, J.Gregory (2003) *Basket Default Swaps, CDO's and Factor Copulas*. Preprint.
- [5] Kalemánova, Schmid, Werner (2005) *The Normal Inverse Gaussian distribution for synthetic CDO pricing*. Preprint.