Adenza

Calypso Credit Derivatives Analytics

Version 17.0

February 2022 – First Edition

Welcome to the Calypso Credit Derivatives Analytics guide which aims to provide an understanding of the analytics that underlie the Calypso Credit Derivatives pricing capability.

Revision Date	Comments
February 2022	First edition for version 17.0.

Contents

Section 1. Pricing Credit Derivatives - Overview 7

1.1 Products 7

1.2 Pricers 7

Section 2. Valuation of Credit-Contingent Cashflows 11

2.1 Characterization of Credit-Contingent Cashflows 11

2.2 Valuation under Deterministic Probabilities and Rates 11

2.3 Valuation Formula 12

2.4 Summation Approximations 12

Section 3. Probability Curve Generation 14

3.1 Inputs 14

3.2 Iterative Procedure 14

3.3 Interpolation Methods 15

Section 4. Pricing of Credit Default Swaps 17

4.1 Definitions 17

4.2 Valuation of the Protection Leg 17

4.2.1 Credit Default Payment Types 18

4.2.2 Lookback Period 19

4.3 Valuation of the Premium Leg 19

4.3.1 No Accrual 20

4.3.2 Full Coupon 20

4.3.3 Riskless 20

4.3.4 Single Payment 20

4.3.5 Pay Accrual 20

4.4 Calculation of the Total NPV and Breakeven CDS Spread 22

4.5 Analytical Benchmarks 22

4.5.1 Analytic Evaluation of the Protection Leg 23

4.5.2 Analytic Evaluation of the Premium Leg 23

4.5.3 Total NPV for Sampling on Premium Dates 24

4.5.4 Closed-Form Breakeven Rates 24

4.5.5 Analytic Benchmark Examples 25

Section 5. Pricing of CDS Baskets 26

5.1 Adding New Pricers and Models 26

5.2 Pricer Measures 26

5.3 Termination Payment Parameters (Credit Protection Leg) 32
5.4 Premium Payment Parameters (Premium Leg) 32
5.5 Pricer Parameters: CDS Baskets 33
Section 6. Monte Carlo Evaluation of CDS Basket Swaps 36
6.1 Procedure 36
6.1.1 CDSNthLoss (CDO Tranche) Evaluation 36
6.2 Implementation....37
6.2.1 Storage of Random Numbers 38
Section 7. General Tests and Performance: Monte Carlo.... 40
7.1 Performance 40
7.2 Example Values....40
7.3 Relationships Among CDS Basket Values 41
7.4 Analytic Examples....43
Section 8. One-Factor Model 44
8.1 One-Factor Model Loss Distribution....44
8.1.1 Correlation Modeling....44
8.1.2 Loss Distribution Calculation....46
8.2 Calculating NPV from the Loss Distribution 47
8.3 Non-Homogeneous Small Basket Nth Default Model 48
8.4 Recommended Parameter Settings: Speed Versus Accuracy 50
8.5 Calypso Classes: How to Modify or Add Credit Models 52
Section 9. CDS Index and Index Tranche Pricing.... 55
9.1 Valuation from Probability Curves (0-100% Tranche)....55
9.2 Valuation from Clean Price Quote (Price-Based Index)....56
9.3 Valuation from Index Spread Quote (CDSW Pricing) 57
9.4 CDS Index Basis Adjustment....66
9.5 CDS Index Tranche Valuation 67
Section 10. Base Correlation Surfaces 68
10.1 Base Correlations and Correlation Skew 68
10.2 Generation of Base Correlations 68
10.2.1 Market Data Inputs....68
10.2.2 Pricing a Tranche from Base Correlations 69
10.2.3 Generating Base Correlations from Breakeven Prices....70
10.2.4 Super-Senior Tranches....70
10.3 Extrapolation in Time 71
Section 11. Bespoke Basket Correlations.... 72

11.1 Introduction to Bespoke Mapping 72
11.2 Expected Loss Method....72
11.3 Moneyness Method....74
11.4 Mapping from Multiple Surfaces: Correlation Formula....74
11.5 Usage: Pre-Generation Versus Mapping on Demand 75
11.5.1 Single Surface....75
11.5.2 Multiple Surfaces....76
Section 12. Pricing of Funded CDS and CDS Baskets 77
12.1 Single-Name CDS....77
12.1.1 NPV of Funded CDS 77
12.1.2 Breakeven Rate of a Funded CDS: Price from Curve 78
12.2 Funded CDS Baskets 78
12.2.1 NPV of Funded CDS Basket....79
12.2.2 Breakeven Rate of Funded CDS Basket....79
12.2.3 Funded CDS Index Price from Quote 79
Section 13. Senior Amortization and Incurred Recovery 81
13.1 Calculation of Incurred Recovery and Incurred Loss 81
13.2 Example 82
Section 14. Random Recovery Model.... 85
14.1 Crisis and Recovery 85
14.2 RECOVERY_MODEL Parameter....86
Section 15. CDO Squares.... 87
Section 16. Credit Default Swaptions 88
16.1 Expiration Payout of Single-Name Swaption 88
16.2 Valuation of Single-Name Swaptions....89
16.3 Pricing Formula 89
16.4 Required Market Data 90
16.4.1 CDS Volatility Surfaces 90
16.4.2 CDS Market Data....91
Section 17. CDS Index Options.... 92
17.1 Generalization for Index Basket Options....92
17.2 Cash Settlement of CDS Index Options....92
17.3 Pre-Expiry Default Protection 92
17.4 Arbitrage Free Credit Index Option Valuation 94
17.4.1 Option PayOff....94
17.4.2 Option Valuation....96

17.4.3 Implementation 97
Section 18. Constant Maturity CDS (CMCDS) 98
18.1 CMCDS Rate Index 98
18.2 CMCDS Pricer 98
18.3 Convexity Adjustment 99
Section 19. CDS on ABS/ABX 100
19.1 Pricers 100
19.2 Valuation using Market Quotes for Spread and WAL 100
Section 20. Credit-Linked Contracts 102
20.1 Credit-Linked Notes (CLNs) 102
20.1.1 Definition 102
20.1.2 Calypso CLN Bond and Pricer Configuration 102
20.1.3 Valuation by CDS Replication 103
20.1.4 CLN Issuer Risk 104
20.2 Credit Contingent Interest-Rate Swap 104
20.2.1 Definition 105
20.2.2 Valuation by CDS Replication 105
20.2.3 Credit-Contingent Single Swap Legs 105
20.2.4 Pricers 106
Section 21. Performance Swap 107
21.1 Valuation 107
21.1.1 Forward Price Projection 107
21.1.2 Expected Value Computation 107
21.2 Pricers 108
21.2.1 PricerPerformanceSwap 108
21.2.2 PricerPerformanceSwapAccrual 108
Appendix 109

Section 1. Pricing Credit Derivatives - Overview

1.1 Products

A CreditDefaultSwap provides credit protection or exposure to a single reference entity (an issuer or asset). To trade credit exposure on multiple reference entities, the products CDSNthDefault and CDSNthLoss are used. These capture the concepts of credit default swaps (CDS's) and credit default obligations (CDO's) on baskets of assets. As with the single-name product, the basket products have a credit protection leg and a premium (interest rate swap) leg.

The CDSNthDefault defines protection according to the order in which default occurs. Parameters on the trade define whether credit protection payments occur on a single default - first to default, second to default, etc. – or on a range of defaults, such as 2nd through 4th. When the largest specified number of defaults has occurred, the trade terminates.

The CDSNthLoss defines protection in terms of monetary amounts. Minimum and maximum loss amounts are specified at which credit protection payments begin and end. The trade terminates when the maximum loss has occurred. The product can be used to represent a CDO tranche or an index.

As each default occurs, the amount of remaining protection decreases and the notional of the premium payments is reduced accordingly.

The following features of these trades are not found on ordinary interest rate swaps:

- Basket definitions
- Probability curve assignment for each asset in the basket
- Correlation matrix definition for the returns of each asset
- Termination payment parameters on the credit protection leg
- Payment parameters on premium leg
- Credit events in trade lifecycle define default events and alter future payments

Please refer to the user documentation for Trade Worksheets for specific information on entering the trades and the trade lifecycle credit events.

1.2 Pricers

The Calypso pricers provide functionality that is used by all valuation models, such as the obtaining of market data and handling the mechanics of empirical risk measures. The computation of NPV, however, is performed by a model class separate from each pricer. In many cases, the only difference between pricers is the model that is called on to compute NPV.

The following pricers and models are available for pricing Credit Derivatives. Details of the models are described in subsequent chapters.

Pricer	CDS Type	Valuation Model
CreditDefaultSwap	Single Name CDS	Deterministic Default Probability.
CreditDefaultSwapRisky	Single Name CDS with Protection Seller Risk	Gaussian Copula.

Pricer	CDS Type	Valuation Model
CreditDefaultSwapFlat	Single Name CDS	Price from Quote, ISDA Model. Interpolates across probability curve's underlying quotes based on generated end dates vs. trade maturity dates. Resulting quote set as BE_RATE.
CreditDefaultSwapQuote	Single Name CDS	Price from Quote, ISDA Model. BE_RATE is obtained from the quote set.
CDSNthDefault	Nth-to-default	Generic class, defaults to Monte Carlo.
CDSNthDefaultOFM	Nth-to-default	One factor model, Gaussian copula.
CDSNthLoss	Tranche	Generic class, defaults to Monte Carlo.
CDSNthLossMC	Tranche	Monte Carlo, Gaussian copula.
CDSNthLossOFM	Tranche	One factor model, Gaussian copula.
CDSNthLossBespoke	Tranche	Multiple correlation surface version of CDSNthLossOFM combined with expected loss mapping.
CDSNthLossOFMHermite	Tranche	Fast pricer using Hermite (Gram-Charlier) expansion to approximate the Bernoulli distributions for the one-factor Gaussian copula model.
CDSNthLossRR	Tranche	Random Recovery extension of the Gaussian Copula.
CDSIndex	Index	(1) Component evaluation (basis adjusted). (2) Price from market quote.
CDSIndexTranche	Index tranche	Models tranche as an NthLoss product, then users whichever pricer is assigned to NthLoss in the PricerConfig.
CDSIndexTrancheOFM	Index tranche	Models tranche as an NthLoss product, which it prices with CDSNthLossOFM.

Pricer	CDS Type	Valuation Model
CDSIndexTrancheRR	Index tranche	Random Recovery extension of the Gaussian Copula.
CreditDefaultSwaption	Option on CDS	Black-type model where the underlying CDS is valued using the pricer assigned to that product in the PricerConfig.
CDSIndexOption	Index option	Black-type model with pre-expiry protection, with the underlying valued with pricer assigned to NthLoss trades in the PricerConfig.
CDSIndexOptionMC	Index option	CDSIndexOption pricer with the underlying valued as a whole-basket (100% tranche) NthLoss product using CDSNthLossMC.
CDSIndexOptionOFM	Index option	CDSIndexOption pricer with the underlying valued with the CDSIndex pricer, which gives the same result as valuing a whole-basket (100% tranche) NthLoss product using CDSNthLossOFM.
CDSIndexTrancheOption	Index tranche	CDSIndexOption pricer with the underlying valued with pricer assigned to NthLoss trades in the PricerConfig.
CDSIndexTrancheOptionMC	Index tranche	CDSIndexOption pricer with the underlying valued as an NthLoss tranche using CDSNthLossMC.
CDSIndexTrancheOptionOFM	Index tranche	CDSIndexOption pricer with the underlying valued as an NthLoss tranche using CDSNthLossOFM.
CreditDefaultSwapABS	CDS on ABS	Prices CDS on ABS from market quotes for spread and weighted average life.
CDSABSIndex	CDS on ABX Index	Prices CDS on ABX index from market quotes for spread or price.
CDSABSIndexTranche	CDS on ABS Index tranche	Prices CDS on ABX index tranche from market quotes for price.

Pricer	CDS Type	Valuation Model
CMCDS	Constant-maturity CDS	CMCDS pricing with projection of forward constant-maturity CDS reset rates with convexity correction.

Credit-related structured products have pricers that delegate their computations to one of the CDS pricers listed above or to the associated valuation model. These pricers include the following.

Credit-Contingent Pricer	Product and Subtype
PerformanceSwap	Performance Swap
PerformanceSwapAccrual	Performance Swap
SwapCreditContingent	Swap - Credit Contingent
SwapCreditContBasket	Swap - Credit Contingent Basket
SingleSwapLegCreditContingent	Single Swap Leg - Credit Contingent
SingleSwapLegCreditContBasket	Single Swap Leg - Credit Contingent Basket
BondCLN	Bond CLN - Standard
BondCLNBasket	Bond CLN - Basket

Section 2. Valuation of Credit-Contingent Cashflows

2.1 Characterization of Credit-Contingent Cashflows

Trades that are exposed to credit risk can typically be decomposed into *credit-contingent cashflows*, payments which have some deterministic dependence on the credit events -- default, bankruptcy, etc. -- of one or more reference entities. Then the value of the trade can be found by summing the present expected values of the cashflows under some model of credit event probabilities and interest rates.

A credit-contingent cashflow will pay either (a) an amount *A* on a scheduled date if there is no credit event, or (b) an amount *C(t)* if there is a credit event at time t, where t falls within a credit contingency period between times t_1 and t_2, $t_1 <= t < t_2$.

The payment amounts A and C(t) may either be fixed amounts or dependent on a floating-rate index or additional non-credit market events.

2.2 Valuation under Deterministic Probabilities and Rates

To find the present expected value of a credit-contingent cashflow one must have an estimate of the probability of default (and any other triggering credit event) as a function of time, as well as the discount factors for risk-free payments. The simplest model is to assume the probabilities and risk-free discount factors are known and will not change prior to default. This is the standard assumption used in the marketplace.

Suppose then that one has determined the cumulative probability of default for any future time. Let this be p(t). Thus if t_E is the default event,

$$p(t) = \Pr(0 <= t_E < t)$$

The probability that the reference asset survives without default to time t is q(t) = 1 - p(t).

Note: In Calypso, the probability curve is the survival probability q(t) rather than the default probability p(t). So, e.g., Calypso curves begin at q(0) = 1, rather than the default probability p(0) = 0.

From this expression, the probability that a credit event occurs between two future times t_a and t_b is

$$\Pr(t_a <= t_E < t_b) = p(t_b) - p(t_a)$$

The probability of no credit event in this period is just this value subtracted from one.

The *expected value* of the contingent cashflow is the contingent payment amount on any date weighted by the probability that amount is paid, summed over each day in which a contingent payment is possible. The *present expected value* is the expected value discounted by the risk-free factor D(t).

For the regularly scheduled payment A payable at maturity T_M if there is no credit event between t_1 and t_2 the present expected value is

$$(1 - p(t_2) + p(t_1))AD(T_M)$$

For a contingent payment of value C(ta) if default occurs on day ta -- that is, between the start of day on t_a and the start of day on the next day, t_a + 1 -- the present expected value is

$$(p(t_a + 1) - p(t_a))C(t_a)D(t_a)$$

In these expressions an allowance can be made for the lag between the occurrence of the default event and the actual receipt of the corresponding payment. This lag will depend on the terms of the settlement agreement, and D(t) can be defined appropriately. Let $\overline{D}(t)$ be the actual discount factor for discounting a flow at time t, without a lag. Then D(t) can be defined for the agreed or assumed settlement conditions as follows:

- Immediate settlement upon default: $D(t) = \overline{D}(t)$
- Settlement with a time lag of L: $D(t) = \overline{D}(t+L)$
- Settlement delayed until normal maturity: $D(t) = \overline{D}(T_M)$

And for the regularly scheduled payment one always has $D(T_M) = \overline{D}(T_M)$.

2.3 Valuation Formula

The present expected value of a contingent cashflow exposed to credit risk between t_1 and t_2, paying at T_M, is

$$V = (1 - p(t_2) + p_0)AD(T_M) + \sum_{i=0}^{N} C(T_i)D(T_i)(p(T_{i+1}) - p(T_i))$$

assuming deterministic probabilities and interest rates. Here:

i: index of each day in the credit-sensitive lifetime of the cashflow, i = 0, 1, ... N, such that T_0= t_1 and T_N= t_2.

A: Scheduled amount if no credit event

$C(T_i)$: Amount if there is a credit event on day T_i

$D(T_i)$: Function identifying the discount factor for the settlement payment associated with day T_i. This function may include a fixed settlement lag added to T_{i+1} or a schedule of allowed settlement dates, depending on the context.

$p(T_i)$: Probability that a credit event has occurred prior to day T_i

p_0: Typically equal to zero, this is the probability on the value date that a credit event has occurred prior to the contractual start date of the cashflow, assuming the cashflow is nullified if an event occurs before the start date (A = C = 0 in that event).

2.4 Summation Approximations

The C(t) term is complicated by the summation over time. There is time dependence in the discount factor D(t) and the probabilities p(t) as well as the payment amount C(t). A typical example of time-dependent payment is an interest cashflow which, upon default, pays out the amount of interest accrued during their interest period up to the time of default (a common type for the premium cashflow for credit default swaps). Another example is a protection payment where the forecasted recovery value is not constant (expressed as a time-dependent recovery rate).

A daily summation of these time dependent values is computationally costly, but can be made much faster by good approximations. If there are analytic expressions for D(t) and p(t), such as linear interpolation between a small number of points, the summation can be done analytically. More generally, one can approximate the summation as an integral

$$I = \int_{t_1}^{t_2} \frac{dp(t)}{dt} C(t) D(t) dt$$

and use standard numerical integration methods to speed computation. Two effective methods in this case are the trapezoid rule

$$\int_{t_1}^{t_2} f(t)dt = \frac{1}{2}(f(t_1) + f(t_2))(t_2 - t_1)$$

and Simpson's rule

$$\int_{t_1}^{t_2} f(t)dt = \frac{(t_2 - t_1)}{6}(f(t_1) + 4f(\frac{t_1 + t_2}{2}) + f(t_2))$$

These rules can also be applied to sub-periods comprised of regions in which f(t) is relatively smooth.

A common approximation in the market place is J P Morgan's approach, which makes two assumptions: (1) the probability curve is linear over the cashflow period, and (2) the payment is always made on the regular cashflow payment date. In that case, the density *dp/dt* is just the constant slope between the start and end points of the flow, and the discount factor is just the factor on the payment date. The integral becomes, for time-independent C(t),

$$\sum_{i=0}^{N} C(T_i) D(T_i)(p(T_{i+1}) - p(T_i)) \approx CD(T_N)(p(T_N) - p(T_0))$$

This has the virtue of being a fast calculation, but the inaccuracy can be significant if there are rapid changes in the slope of p(t) within the summation period.

In Calypso, the type of integration approximation can be chosen using the model parameter INTG_METHOD_CR.

Section 3. Probability Curve Generation

A Calypso probability curve represents the probabilities that a reference asset will survive to any given time. The CurveProbability application will generate a curve from market spreads for Credit Default Swaps on the reference asset. Alternatively, the application can be used to manually create a curve with externally-generated probabilities.

3.1 Inputs

The generation from market spreads is performed using an iterative search. The method is very similar to that employed by Calypso for zero curve generation. The inputs are as follows:

- The user creates, as needed, Curve Underlyings representing Credit Default Swaps on a reference asset.
- The user selects the desired CDS Curve Underlyings. Typically these will all be on the same reference asset but represent different tenors. The user enters the market spreads for each tenor.

  Adding Bonds: In addition to CDS, Bonds can be chosen as underlyings and their market prices entered on the quotes tab.
- A riskless zero curve is selected.
- A recovery rate is entered.
- An interpolator is chosen. Currently supported are Linear and Log-Linear interpolation. (See below.)

3.2 Iterative Procedure

Then generation proceeds as follows:

1. A curve is begun with a single point on the curve generation date, with survival probability 1 (100%).
2. For the Curve Underlying with the shortest tenor, a Credit Default Swap trade is created with the quoted spread as the premium rate.
3. A guess is made for the probability as of the last date of the CDS. This point is added to the curve. The probabilities between this point and the previous point are therefore implicitly defined by the chosen interpolation method. This creates a candidate curve.
4. The CDS is priced with the candidate curve to find the Net Present Value. The specified zero curve and recovery rate are also used in this calculation.
5. From the calculated Net Present Value, an improved guess of the survival probability is made.
6. Steps 3 – 5 are repeated until a candidate curve produces a zero Net Present Value. More exactly, the computation halts when change in the probability cannot be found that produces a more than 10^{-14} improvement in the Net Present Value.
7. The CDS Curve Underlying with the next longer tenor is now used and the above steps repeated, beginning with Step 2.
8. The curve is completed when the longest CDS Curve Underling has been used.

To compute the Net Present Value the generation will use the pricer specified for Credit Default Swaps in the PricerConfig. The default pricer to calculate this value is PricerCreditDefaultSwap. If the user has a custom pricer configured instead, the probability curve generator will employ it automatically.

Solving with Bonds: If Bonds are used in addition to CDS the same procedure is applied. Pricing is done using the appropriate bond pricer, the discount curve and the candidate probability curve, so as to reproduce the market value of the bond.

3.3 Interpolation Methods

Given survival probabilities at times t_1 and t_2, interpolation produces the probability at an intervening time t, $t_1 <= t <= t_2$. Define:

$$t = \alpha t_1 + \beta t_2,$$
$$\alpha = \frac{t_2 - t}{t_2 - t_1}, \quad \beta = 1 - \alpha.$$

Two interpolation methods are currently available.

***Linear Interpolation*:**

$$p(t) = \alpha p(t_1) + \beta p(t_2)$$

***Log-Linear Interpolation or Constant Hazard Rate Interpolation*:**

This is linear interpolation on the logarithms of the probabilities

$$\ln p(t) = \alpha \ln p(t_1) + \beta \ln p(t_2),$$

which solves to

$$p(t) = p(t_1)^{\alpha} p(t_2)^{\beta}.$$

The Log-Linear method is identical to *Constant Hazard Rate Interpolation*, which introduces a constant instantaneous hazard rate *h* between t_1 and t_2 defined by

$$h = \frac{-\ln(p(t_2)/p(t_1))}{t_2 - t_1},$$
$$p(t_2) = p(t_1)e^{-h(t_2 - t_1)}$$

One sees the equivalence of the two methods as follows. In Constant Hazard Rate Interpolation, the interpolated probability at any t between t_1 and t_2 is defined to be

$$p(t) = p(t_1)e^{-h(t - t_1)}$$

Substituting the definition of *h* produces

$$
\begin{aligned}
p(t) &= p(t_1)e^{\frac{\ln(p(t_2)/p(t_1))}{t_2-t_1}(t-t_1)} \\
&= p(t_1)(p(t_2)/p(t_1))^{\frac{(t-t_1)}{t_2-t_1}} \\
&= p(t_1)(p(t_2)/p(t_1))^{\beta} \\
&= p(t_1)^{1-\beta}\, p(t_2)^{\beta}
\end{aligned}
$$

which is the Log-Linear method.

In other terminology, this can be called interpolation with *piecewise-linear cumulative* hazard rates, where the cumulative hazard function *H(t)* is the integral of the instantaneous hazard rate *h(t)*:

$$p(t) = \exp(-\int_0^t h(t')dt') = \exp(-H(t))$$

Section 4. Pricing of Credit Default Swaps

A buyer of protection in a single-name credit default swap (CDS) will make regular premium payments at a given rate, the CDS spread, in exchange for contingent payments in the event of the default of a reference asset. There are several forms of the premium and the default protection payments.

Both legs of the CDS are composed of credit-contingent cashflows that can be valued under deterministic probabilities and interest rates using the methods described in the chapter "Valuation of Credit-Contingent Cashflows." The formula for the present expected value of one such cashflow was given to be:

$$V = (1 - p(t_2) + p_0)AD(T_M) + \sum_{i=0}^{N} C(T_i)D(T_i)(p(T_{i+1}) - p(T_i))$$

The net present value of the CDS is just the sum of the present expected values of the cashflows. The present chapter describes the credit-contingent calculations applied to the special case of the CDS.

These methods are implemented in the Calypso class `PricerCreditDefaultSwap`.

References

Credit, Arvanitis and Gregory, (Risk Books, 2001)

2003 ISDA Credit Derivatives Definitions (ISDA, 2003)

"Credit Derivatives: A Primer" (JP Morgan, Jan 2005)

4.1 Definitions

The following definitions will be used:

- D(t): discount factor for discounting premium payment made at time t
- $D_c(t)$: discount factor for discounting default payment made at time t
- S(t): the probability that the asset will survive without default to time t
- I: notional principal of the CDS; can be amortizing
- t_k: End of the kth premium accrual period
- t_k^P: Premium payment date for the kth premium accrual period
- t_0: The value date or the start date of the CDS, whichever is later

4.2 Valuation of the Protection Leg

The protection leg can be characterized as a single credit-contingent cashflow which pays nothing if there is no default, and pays some amount C in the event of default. The value today of the payment is the discounted expected value

$$NPV_{protection} = E[D_c(t)C]$$

where the expectation value *E* is taken using the survival probabilities. Using the valuation formula for credit-contingent cashflows given above, one has

$$NPV_{protection} = \sum_{i=0}^{N} C(T_i)D(T_i)(p(T_{i+1}) - p(T_i))$$

To compute this value, one must use one of the approximations described previously. The Calypso parameter INTG_MTHD_CR controls the choice of approximation. In the rest of this chapter the formulas for the choice LINEAR_SINGLE are described. In this method, a sampling period is used that is typically equal to the premium payment period, and the discounting is performed assuming payment is made at the end of the sample period, with a possible time lag.

If payment is made immediately on default, then with sample period τ and survival probability S(t) = 1 - p(t),

$$NPV_{protection} = \sum_{i=1}^{N} D_c(i\tau)(S((i-1)\tau) - S(i\tau))C \text{ (no payment delay)}$$

A delay in payment can be taken into account: if there is an average lag of m days between the credit event and the payment date, then the NPV is multiplied by an additional factor of exp(-zm/365).

A CDS that "Settles at Maturity" will delay its protection payment (if any) until the regular maturity date of the CDS. In this case, the discount factor is just the factor for that date, no matter the sample period.

As mentioned, the sampling for the LINEAR_SINGLE method is by default the premium accrual dates for the sampling dates of the credit leg. If there are no accrual dates, such as is the case of a single up-front premium payment, a 90-day sampling period is used.

Setting INTG_MTHD_CR to ANALYTIC_JPM implements the ISDA methodology for standard coupon CDS. The method sums up the expected payments using analytic interpolation on the given curves; for the discount curve it uses piecewise constant forward rates. A user should employ the BootstrapISDA generator in the zero curve window together with the ISDA-approved quotes.

The ANALYTIC_JPM method converts between spread and upfront and computes the cash settlement amount. The upfront is clean quoted upfront on T. The cash settlement amount is paid on buyer protection to protection seller on T+3 Business Days.

Where:

- The accrued is on the coupon between the accrual date and T (includes T).
- The clean price is 100-upfront. The inputted maturity date would have the 20th of the month assumed.
- The standard coupon would be either 100bps or 500bps.

ISDA specifies the recovery rate be 40% for SENIOR and 20% for UNSUBORDINATED.

The spread will be calculated if the upfront fee is inputted and vice versa. The cash settlement amount and cash settlement date will also be generated. The Constant Hazard Rate is solved for giving a standard CDS a MTM equal to notional*upfront - accrued premium discounted riskless to T. Solving for the spread will give a CDS with a MTM equal to minus its accrued premium discounted riskless to T. The Constant Hazard Rate is then applied to calculate the Cash Settlement Amount, the Upfront and the Spread.

- Cash Settlement Amount - MTM of the standard CDS / riskless discount factor to T + 3 business days.
- Upfront - (cash settlement - accrued premium) / notional.
- Spread - standard CDS except that Coupon Rate is equal to the spread that is solved for. Solving for the Coupon Rate gives an Upfront of 0.

4.2.1 Credit Default Payment Types

Evaluating the NPV depends on the payment sampling rate and the definition of the termination payments.

Calypso employs the common practice of

The following are forms the credit default payment can take.

Par minus Recovery: The payment is equal to the fraction of notional principal lost on default. If *R* is the realized recovery rate and *I* the CDS notional, then

$$C = (1 - R)I$$

In the case of an amortizing premium leg the notional *I* is a known function of time and the default payment depends upon the notional at the time of the credit default event. The recovery rate is unknown until payment. In the present model, a fixed value of the rate is estimated in order to evaluate the NPV.

Initial minus Recovery: The payment is an agreed-upon initial value of the asset, less the recovered principal. If *w* is the agreed initial value per unit notional, then

$$C = (w - R)I$$

Fixed amount: C is an amount agreed upon at the inception of the trade.

Fixed percentage: C is a percentage of notional agreed upon at the inception of the trade; if the notional is amortized (as defined on the premium leg), the payment is that percentage of the notional as of the time of the default event.

4.2.2 Lookback Period

In accordance with the ISDA rules, the single-name credit default swap includes a lookback period for application of credit defaults. In particular, a trade entered today will have exposure to defaults that can occur later today, even though the trade start date is the next business day – because today is within the lookback period.

The lookback period is defined in domain "creditEventLookBackTenor" in the form of a tenor like 2M, 14D, etc.

4.3 Valuation of the Premium Leg

The premium leg comprises cashflows which pay a scheduled payment A_k, for the *kth* cashflow, if no default has occurred prior to the end of the *kth* premium period, and a payment B_k if default has occurred -- while B_k can be zero, the most typical CDS make a payment equal to the accrued amount of the premium as of the date of default. The valuation formula for credit-contingent flows can again be applied.

Using the LINEAR_SINGLE approximation to the summation, sampling is done only on the start and end dates, and the formula can be written in a simple form. Let date t_{k-} denote the start of the kth premium period and t_k, the end date, with the payment itself occurring at t_k^P. The expected present value of the premium leg is then

$$NPV_{premium} = \sum_{k=1}^{M} D(t_k{}^P)[S(t_k)A_k + (S(t_{k-1}) - S(t_k))B_k]$$

In this formula, the values A_k and B_k are negative if the premium is to be paid out, and positive if premium is received.

CDS trades can have a variety of definitions for the A and B payments. The most common premium is fixed rate, $A_k = rIa_k$, with *r* the fixed rate, that is, the CDS spread; *I* the notional of the CDS; and a_k the accrual period for the kth payment date. Note: For the last payment, the accrual includes the CDS trade's maturity date and the payment can fall after, but in this case the NPV formula the probability $S(t_k)$ is still taken at the CDS maturity date.

There are several common definitions for the B_k payments upon default. These are described in the following sections.

Using the ANALYTICS_JPM for summation of the premium leg assumes the accrual beginning date as the first adjusted CDS date on or before T+1 where the adjusted dates business day adjusted Following. The premium accrued before and including T is considered riskless. This is the number of days = T - accrual begin date + 1.

The premium accrued after and including T+1 is considered risky, where the number of days = maturity date - T = number of days of protection.

4.3.1 No Accrual

No payments are made after default. The asset must survive to t_k in order for A_k to be paid. In the NPV formula, B_k=0.

$$NPV_{premium} = \sum_{k=1}^{M} D(t_k{}^P)S(t_k)A_k$$

4.3.2 Full Coupon

One payment is made after default: the full regular premium payment for the period in which the default occurs. In pricing, one uses the fact that a coupon A_k is paid if the asset has survived to the start t_{k-1} of the premium period.

$$NPV_{premium} = \sum_{k=1}^{M} D(t_k{}^P)S(t_{k-1})A_k$$

4.3.3 Riskless

All premium payments are made regardless of the credit event. There is no dependence on the survival probability.

$$NPV_{premium} = \sum_{k=1}^{M} D(t_k{}^P)A_k$$

4.3.4 Single Payment

Only one guaranteed payment is made.

$$NPV_{premium} = D(t_0{}^P)A_0$$

4.3.5 Pay Accrual

When default occurs, the interest that has accrued up to the time of the default is paid at the next premium payment date.

As the amount of the payment depends on the time at which default occurs, the sampling of accrual periods needs to be more frequent than the regular coupon frequency in order to find an accurate expected value. The LINEAR_SINGLE method may not be accurate enough, in which case the SIMPSON choice for INTG_MTHD_CR is recommended. LINEAR_SINGLE the simplest and fastest estimate, and corresponds to that used in the J P Morgan (Bloomberg "J") model of CDS valuation, while the SIMPSON choice approximates the Hull-White method (Bloomberg "H").

The LINEAR_SINGLE (Morgan) method estimates accrual by observing that default will occur on average in the middle of the period. Then $B_k = A_k / 2$ for k > 1. For k=1, i.e., the current premium period in which the valuation date falls, some accruall has already occurred and only the remaining portion is at risk, so a default is supposed on average to occur in the middle of the remaining period. That is, *f* is the fraction of the current period that has accrued so far, with 0<= *f* <= 1, then $B_1 = fA_1 + (1-f)A_1 / 2$. Then one can write the NPV formula as:

$$NPV_{premium} = \sum_{k=1}^{M} D(t_k{}^P)[S(t_k)A_k + (S(t_{k-1}) - S(t_k))(1 + f\delta_{k1})A_k / 2] \text{ (Pay accrual)}$$

with

t_0 = the value date or the start date of the CDS, whichever is later;

f = fraction of first period accrued as of t_0;

δ_{k1} = 1 if k = 1, and 0 otherwise.

When using the SIMPSON method, the accrual A(t) as a function of time is employed in the Simpson's Rule approximation of the integral:

$$NPV_{premium} = D(t_1{}^P)[A_1(t_{val}) + S(t_1)(A_1 - A_1(t_{val})) + \int_{t_{val}}^{t_1} (-\frac{\partial S(t)}{\partial t})(A(t) - A_1(t_{val}))dt]$$

$$+ \sum_{k=2}^{M} D(t_k{}^P)[\delta_{k1} + S(t_k)A_k + \int_{t_{k-1}}^{t_k} (-\frac{\partial S(t)}{\partial t})A(t)dt]$$

This expression separates the first, currently-accruing period from the forward-starting periods. Using Simpson's Rule together with the interpolation method defined on the probability curves, the integrals are readily calculated.

Relation between the Approximation Methods

LINEAR_SINGLE valuation can be viewed as a simplification of the integral evaluation by approximating the survival probability *S(t)* as a linear function over each premium period. The slope is determined by the endpoints of the period:

$$h_k = -\frac{S(t_k) - S(tk_{i-1})}{(t_k - t_{k-1})} > 0$$

$$S(t) = S(t_{k-1}) - h_k(t - t_{k-1}), \quad t_{k-1} <= t <= t_k$$

Then one has:

$$\int_{t_{k-1}}^{t_k} \left(-\frac{\partial p(t)}{\partial t}\right) A_k(t)dt = \int_{t_{k-1}}^{t_k} h_k Nc(t - t_{k-1})dt$$
$$= h_k \frac{(t_k - t_{k-1})^2}{2} Nc = (S(t_k) - S(t_{k-1})) \frac{(t_k - t_{k-1})}{2} Nc$$
$$= \frac{1}{2}(S(t_k) - S(t_{k-1})) A_k(t_k)$$

As expected, this is the expected value of the premium assuming on average that default occurs at the midpoint of the period.

Changes from Prior Versions of Calypso

In earlier versions of Calypso, the accrual methods were implemented using the parameter ACC_MODEL, which was only applied to PAY ACCRUAL swaps and employed a different numerical integration. This has been replaced by the parameter INTG_MTHD_CR, which applies to all swaps.

Also, the parameter USE_START_PROBABILITY is now obsolete. In earlier versions it was used to replicate the payment type which is now handled by the FULL_COUPON choice.

4.4 Calculation of the Total NPV and Breakeven CDS Spread

The total NPV is the sum of the two leg NPVs when each is defined with the appropriate sign. The leg NPV is positive if the payments of that leg are received, negative if paid.

$$NPV = NPV_{protection} + NPV_{premium}$$

The breakeven CDS spread is the premium fixed rate that produces a zero NPV when the legs are summed. Call the breakeven spread s_{BE}. Then, when the NPV of each leg has the appropriate sign,

$$NPV_{protection} + NPV_{premium} * \frac{s_{BE}}{r} = 0$$

The breakeven spread is therefore given by

$$s_{BE} = -\frac{rNPV_{protection}}{NPV_{premium}}$$

4.5 Analytical Benchmarks

The value of a credit default swap can be computed in closed form if the interest rate and probability curves have an analytic description. Define

z: continuously compounded discount rate (zero rate)

h: continuously compounded hazard rate

Then the discount factor at time t relative to time zero is

$$D(t) = \exp(-zt)$$

and the probability that the asset will survive without default to time t is

$$S(t) = \exp(-ht)$$

It will be assumed the same discount factor applies to the premium and credit default protection leg.

4.5.1 Analytic Evaluation of the Protection Leg

With the given forms for the zero curve and survival curve, the expectation value of the credit leg is:

$$
\begin{aligned}
NPV_{protection} &= \sum_{i=1}^{N} D(i\tau)(S((i-1)\tau) - S(i\tau))C \\
&= \sum_{i=1}^{N} \exp(-z\tau i)(\exp(-h\tau(i-1)) - \exp(-h\tau i))C \\
&= C(\exp(h\tau)-1)\sum_{i=1}^{N} \exp(-(z+h)\tau i) \\
&= C(\exp(h\tau)-1)\frac{1-\exp(-N(z+h)\tau)}{\exp((z+h)\tau)-1}
\end{aligned}
$$

In the continuum limit of infinite N and zero τ,

$$NPV_{protection} = \frac{h}{z+h}(1-\exp(-(z+h)T) \qquad \text{(continuous sampling)}$$

with $T = N\tau$ the time to maturity of the CDS trade.

If there is an average lag of m days between the credit event and the payment date, then the NPV is multiplied by an additional factor of exp(-zm/365).

(Use was made of the series summation $\sum_{i=1}^{n} a^i = \frac{a(1-a^n)}{(1-a)}$.)

4.5.2 Analytic Evaluation of the Premium Leg

The NPV can be evaluated in closed form for the analytic D(t) and S(t) if all the premium accrual periods a_k are the same. Set $a_k = a$. Then one finds:

No accrual:

$$
\begin{aligned}
NPV_{premium}(No\ accrual) &= raI\sum_{k=1}^{M}\exp(-(z+h)ak) \\
&= raI\frac{1-\exp(-M(z+h)a)}{\exp((z+h)a)-1}
\end{aligned}
$$

This quantity, $NPV_{premium}(No\ accrual)$, will be used in frequently in the subsequent formulas.

No accrual, using start probability:

$$
\begin{aligned}
NPV_{premium} &= raI\sum_{k=1}^{M}\exp(-(zak + ha(k-1)) \\
&= \exp(ha)NPV_{premium}(No\ accrual)
\end{aligned}
$$

Pay accrual:

$$NPV_{premium} = NPV_{premium}(No\ accrual)(\frac{1}{2}(\exp(ha)+1))$$

Riskless:

$$NPV_{premium} = raI\sum_{k=1}^{M}\exp(-zak)$$
$$= raI\frac{1-\exp(-Mza)}{\exp(za)-1}$$

4.5.3 Total NPV for Sampling on Premium Dates

When the credit leg sampling period is identical to the premium leg accrual periods, N = M, τ = a, the total NPV can be expressed in a simple form, as then the credit leg NPV is proportional to the "No accrual" premium leg NPV.

NPV total, pay premium, no accrual:

$$NPV = (\frac{C}{raI}(\exp(ha)-1)-1)NPV_{premium}(No\ accrual)$$

NPV total, pay premium, pay accrual:

$$NPV = (\frac{C}{raI}(\exp(ha)-1)-(\frac{1}{2}(\exp(ha)+1))NPV_{premium}(No\ accrual)$$

4.5.4 Closed-Form Breakeven Rates

When the protection leg sampling period is identical to the premium leg accrual periods, one has N = M, τ = a; then the breakeven rate has a simple form for non-riskless premium legs:

No accrual:

$$s_{BE} = \frac{C}{aI}(\exp(ha)-1)$$

Pay accrual:

$$s_{BE} = \frac{2C}{aI}\frac{\exp(ha)-1}{\exp(ha)+1}$$

For small *ha* and a par-minus-recovery payment type, the breakeven rate has a simple approximation:

$$s_{BE} \approx h\,(1-R) \quad , \quad ha << 1$$

4.5.5 Analytic Benchmark Examples

This table shows the values of the analytical formulas for the following CDS assumptions:

- 1,000,000 notional
- Fixed rate, par minus recovery, R = 40%
- Pay premium
- Use end probability for premium
- Use premium date sampling for credit

CC rate	Hazard rate	Premium terms	NPV Credit	NPV Premium (1%)	NPV Total	Breakeven Spread
3%	2%	5Y Quarterly, No accrual	52,888.82	(43,963.92)	8,924.90	1.2030%
3%	2%	5Y Quarterly, Pay Accrual	52,888.82	(44,074.11)	8,814.71	1.2000%
3%	2%	5Y Annual, No Accrual	52,292.86	(43,143.06)	9,149.80	1.2121%
3%	2%	5Y Annual, Pay Accrual	52,292.86	(43,578.84)	8,714.02	1.2000%
3%	2%	5Y Annual, No Accrual, Start prob	52,292.86	(42,288.77)	10,004.09	1.2366%

Section 5. Pricing of CDS Baskets

5.1 Adding New Pricers and Models

Developers who wish to add new models can do so while re-using the existing Calypso pricer classes. Valuation models are implemented in tk.model.credit and have the objective of calculating an NPV without being involved with the mechanics of market data, cashflow creation, or the manipulation of other pricer measures.

A new pricer can be created by subclassing PricerCDSNthDefault and simply changing the createModel method to call the new model. PricerCDSNthDefault.

An example of creating a credit model class is shown in the discussion of the one factor model later in this document.

5.2 Pricer Measures

The following describes some of the pricer measures implemented by `PricerCDSNthDefault` and `PricerCDSNthLoss` and their subclasses.

NOTE that several measures require multiple NPV evaluations and therefore can add significantly to the calculation time, particularly if Monte Carlo pricing is used. These include PV01_CREDIT, PV01_CORRELATION, and particularly IMPLIED_CORRELATION.

AGGREGATE_SPREAD

For CDS Basket trades. The sum of the credit spreads of the individual un-defaulted reference names in the basket. Each spread is the single-name CDS break even rate calculated from the probability curve assigned to the name.

AVG_RECOVERY

For CDS Basket trades. The weighted average of the recovery rates of the curves used in pricing. The weight is the notional value of the associated asset name divided by the current notional of the basket, taking into account losses.

AVG_SPREAD

For CDS Basket trades. The weighted average of the credit spreads of the individual un-defaulted reference names in the basket. The weight is the notional value of the name divided by the current notional of the basket, taking into account losses. The un-weighted sum is the AGGREGATE_SPREAD.

BASE_CORR_ATT

For tranche trades only (CDSNthLoss, CDSIndexTranche). The base correlation used for pricing the losses from 0% to the attachment point.

BASE_CORR_DET

For tranche trades only (CDSNthLoss, CDSIndexTranche). The base correlation used for pricing the losses from 0% to the detachment point.

BE_CORRELATION

For CDS Basket trades and Base Correlation surfaces. The constant value of correlation which produces a zero NPV, not including fees. There may not be a solution, as changing the correlation can only alter NPV within a narrow range. (This is due to the fact that both the premium and the credit default legs depend on correlation, so tend to cancel each other out.) If there is no solution, either a 0 or 1 will be returned, whichever produces an NPV closer to zero.

BE_PCT_OF_AGG

For CDS Basket trades. The ratio of the breakeven rate to the aggregate spread, expressed as a percent.

BREAK_EVEN_RATE

For non-option trades, the premium rate (credit spread) that produces a zero NPV for the trade. The NPV can include fees or not, depending on the BE_INCLUDE_FEES parameter. The rate for a spot-starting trade is found if the parameter BE_INCL_ACC is set to False, which gives the rate so that the NPV without accrued premium is zero.

For CDS Options, the BREAK_EVEN_RATE is the at-the-money rate used in the Black formula. For single-name CDS options, this is just the breakeven rate of the forward swap underlying the option. However, for CDS Index options, the rate used in the Black formula is an adjusted rate due to pre-expiry default protection. It is found from modifying the current index spot rate as described in the CDS Index Option section of this document.

CDS_BASIS_ADJ

The value used in adjusting probability curves so that the calculated breakeven spread of a CDS Index matches an index quote. See the documentation on CDS Index evaluation for more information.

DEFAULT_EXPOSURE

The net loss that would occur on a trade if the asset with the largest breakeven rate defaulted immediately. The default exposure is calculated as the change in NPV due to the hypothetical default plus the expected settlement amount received for the default.

$$\text{Default Exposure} = (\text{NPV after default}) - (\text{Current NPV}) + (\text{Default settlement amount})$$

DEFAULT_NPV

(Single-name trades only.) The NPV of the remaining trade after a hypothetical default of the reference entity taking place on the value date. For standard CDS this is zero, but it will be non-zero if coupons or principal are protected from default, as is common in credit-linked notes.

DURATION

The Risky Duration per unit notional. For premium payment end dates t_k with payment dates t_k^P and accrual period a_k, discount factor D, survival probability S, it is defined as:

$$\text{Duration} = \sum_{k=1}^{M} D(t_k^{\ P})S(t_k)a_k$$

EFF_ATT

For tranche trades only (CDSNthLoss, CDSIndexTranche). The current effective attachment point of the tranche, expressed as a percentage of the original notional of the basket. The effective attachment point takes into account the past losses of the trade.

$$\text{Effective attachment \%} = abs\left(\frac{\max(\text{Original tranche attachment - Past loss},0)}{\text{Current basket notional}}\right) * 100$$

EFF_DET

For tranche trades only (CDSNthLoss, CDSIndexTranche). The current effective detachment point of the tranche, expressed as a percentage of the original notional of the basket. The effective detachment point takes into account the past losses of the trade.

$$\text{Effective detachment \%} = abs\left(\frac{\max(\text{Original Tranche detachment - Past loss},0)}{\text{Current basket notional}}\right) * 100$$

EXPECTED_LOSS

For tranche trades only (CDSNthLoss, CDSIndexTranche). The present value of the trade's future expected losses as a percentage of the current basket notional. The present value of future expected losses is calculating by finding the NPV of the trade's credit protection leg, without fees. The current basket notional takes into account past defaults. Note that in contrast to EXPECTED_LOSS_RATIO, this is displayed as a percent.

$$\text{Expected Loss} = \text{abs}\left(\frac{\text{NPV credit protection leg}}{\text{Current basket notional}}\right) * 100$$

EXPECTED_LOSS_RATIO

For tranche trades only (CDSNthLoss, CDSIndexTranche). The present value of the trade's future expected losses as a fraction of the present value of the whole basket expected loss. In contrast to EXPECTED_LOSS, this is *not* displayed as a percent. The expected loss present values are calculated as the NPVs of the credit protection legs of the tranche trade and a whole basket trade.

$$\text{Expected Loss} = \text{abs}\left(\frac{\text{NPV credit protection leg of tranche}}{\text{NPV credit protection leg of current basket}}\right)$$

IMPLIED_CORRELATION

For CDS Basket trades. A representation of a market correlation matrix by means of a single number. A correlation matrix whose off-diagonal elements are all equal to this value produces the same NPV as the market data correlation matrix. Please note this value is solved for iteratively, so can take five to ten times longer than a single NPV evaluation.

LEVERAGE

For tranche trades only (CDSNthLoss, CDSIndexTranche). The PV01_CREDIT per unit notional of the tranche as a fraction of the same quantity on the whole basket. Equivalently, the ratio of the PV01_CREDIT of the trade to the PV01_CREDIT of a 100% tranche (whole basket) trade of the *same* notional.

$$\text{Leverage} = abs\left(\frac{\text{PV01\_Credit of tranche}}{\text{PV01\_Credit of whole basket}} * \frac{\text{Current notional of whole basket}}{\text{Current notional of tranche}}\right)$$

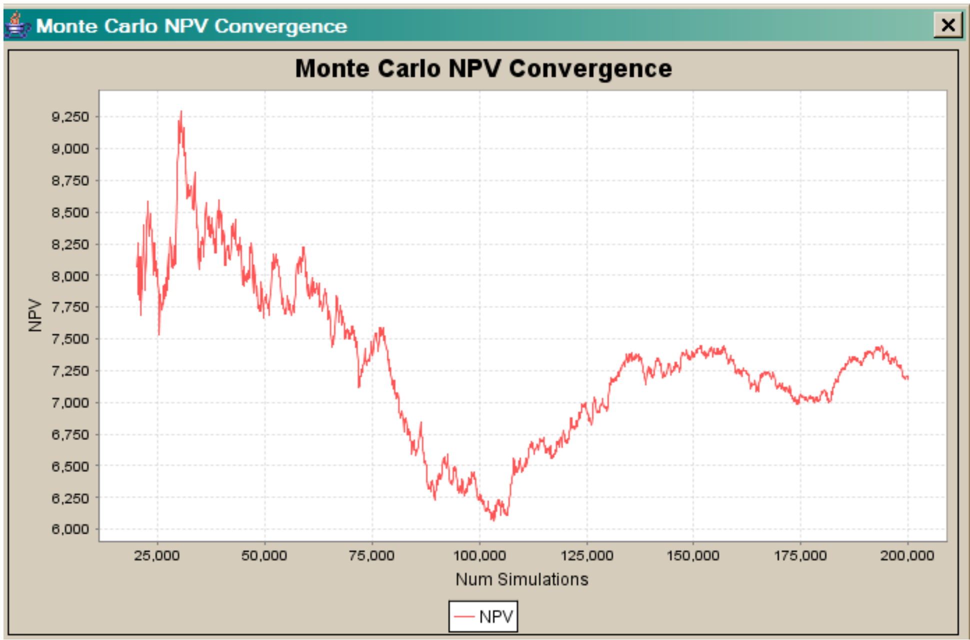
There is an important distinction made in this calculation for index tranches. For CDS Index Tranche trades, the “whole basket” trade used is that of the associated CDS Index. The CDS Index is not quite the same as taking the index tranche trade and setting the tranche to 100%. A trade on a CDS Index will include premium accrued since the last regular coupon payment date, while a CDS Index Tranche only starts accruing premium on the tranche trade's settle date. Thus a "whole basket" trade based on the tranche trade would ordinarily start accruing on the settle date, just as the tranche does; however, in trading one is most interested in hedging tranches with CDS Index trades, not hypothetical whole baskets.

LOSS_GRAPH

A graph of the loss distribution. This pricer measure will display "true" is a graph is available for display; double-clicking on the pricer measure will then open the graph in a pop-up window.

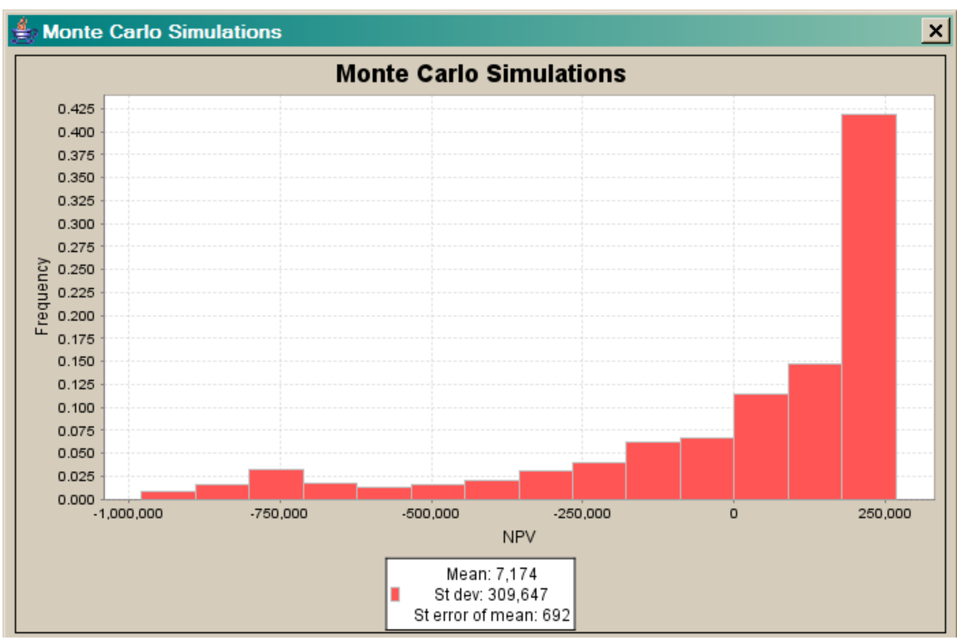
MC_GRAPH

(Monte Carlo Pricers) MC_GRAPH displays the estimate of the NPV for various numbers of Monte Carlo simulations up to the value in the NUM_SCENARIOS parameter. This conveys the convergence of the solution and the residual uncertainty.



MC_HSTGRM

(Monte Carlo Pricers) MC_HSTGRM displays the distribution of the NPV's during Monte Carlo simulation. The Frequency scale shows the fraction of the total number of simulations whose NPV fell within each bin.



NDELTA, NGAMMA

Interest rate sensitivities computed by shifting all CurveZero market data up and down by one basis point, and finding the NPV in each case using the pricer's valuation model. For Monte Carlo, stored random numbers are used to eliminate noise.

NOTIONAL

The current notional amount used on the premium (interest rate) leg. This declines each time there is a default.

NOTIONAL_GROSS

The absolute value of the NOTIONAL measure.

NPV

Computed using the valuation model defined by the pricer. PricerCDSNthLoss will use Monte Carlo, while PricerCDSNthLossOFM will use the Gaussian copula one factor model, and so on. When using Monte Carlo, the pricing parameter USE_STORED_SEED must be set to true to use the same random numbers at each pricing; otherwise, a different NPV will be obtained each time. See also NPV_CREDIT, NPV_PREMIUM.

NPV_CREDIT

The NPV of the credit protection leg.

NPV_PREMIUM

The NPV of the premium payments leg.

PRICE

Same value as NPV.

PV01

Same value as NDELTA.

PV01_CREDIT

Sensitivity to changes in the probability curves. A probability curve is shifted by adding one basis point to the premium rate of the curve's underlying CDS instruments. All of the probability curves are shifted at the same time (there is one curve for each asset in the basket.) With the shifted curves, the NPV is recalculated using the pricer's valuation model. The original NPV is subtracted from this to give the PV01_CREDIT result.

The value can be computed for each name in the basket separately by shifting its corresponding credit curve. To do this, set the parameter PER_REF_NAME to `true`. To view the results, double-click on the PV01_CREDIT field in the trade window. This will bring up a window showing the sensitivity per name and the hedge ratio, i.e., the proportion of the total pv01 credit that is contributed by each name. Please note the additional computation time required for the per-name calculation, as the sensitivity for each name requires a separate NPV evaluation.

PV01_CORRELATION

Change in NPV for 1% change in the attachment and detachment correlations.

PV01_RECOVERY

The change in NPV for a change in PAR_MINUS_RECOVERY protection payments that would result from a 1% absolute change in recovery rate assumptions. This does not compute the effect of a change of recovery on the probability curves. In more detail: Recovery rate assumptions enter into the

NPV of a protection payment in two ways. First, directly through the anticipated payment amount in the event of default:

$$\text{Amount} = \text{Notional} * (1 - \text{RecoveryRate})$$

and second through the probability of default as derived from a market spread, which is roughly

$$\text{Annualized Probability} = \text{Spread} / (1 - \text{RecoveryRate}).$$

The PV01_RECOVERY pricer measure shifts the Recovery Rate in the Amount equation upward by an absolute 1%, that is,

$$\text{NewAmount} = \text{Notional} * (1 - (\text{RecoveryRate} + 0.01)).$$

The NPV is computed for these new amounts, and the PV01 is the new NPV minus the original NPV. The result can be regarded as the credit leg analog of the premium leg's Risky Duration (Spread 01) calculation (see the DURATION pricer measure).

This represents the risk that the actual recovery will not be as anticipated, as opposed to the risk that the market estimate of recovery will change and therefore that quoted CDS spreads will change. To determine the effect of recovery rate changes on the probabilities as well as the protection amounts, one can use the Scenario analysis or the transient RECOVERY_RATE parameter.

We do not recalibrate the survival probability.

5.3 Termination Payment Parameters (Credit Protection Leg)

Cash settlement at the time of a credit default event can be agreed upon in several ways. Each of these are taken into account in the `calculateNPV` method of the valuation model (e.g., MonteCarloCDSBasket). The payment is assumed to occur at the time of default, although in practice it is usually later.

PAR_MINUS_RECOVERY

The payment is equal to the notional value of the defaulting asset in the basket times 1.0 minus the recovery rate. In projecting payments, the recovery rate is obtained either from the asset's probability curve, a recovery curve, or as a manually-set pricer parameter. In addition, the result is multiplied by the Participation parameter on the basket, although this is typically 1.0 (100%).

INITIAL_MINUS_RECOVERY

The user specifies a Reference Price as a percent. The payment is equal to the notional value of the asset times the Reference Price minus the recovery rate.

FIXED_AMOUNT

The user specifies a fixed amount that is paid upon default. Also known as a "binary" payment.

FIXED_PERCENTAGE

The user specifies a fixed percentage. The payment is the notional of the entire basket times this percentage.

5.4 Premium Payment Parameters (Premium Leg)

The behavior of premium payments (interest payments) at the time of a credit default are taken into account in the `calculateNPV` method of the valuation model. The following describes the calculations for the premium payment parameters.

PAY_ACCRUAL

Payment is made for the amount of interest accrued as of the time of default. The calculation approximates this by simple interpolation.

NO_ACCRUAL

No payment is made if default occurs during the interest period.

RISKLESS

All payments are made regardless of default.

Single Premium Payment (Checkbox)

Only a single payment is made on the date specified.

Guaranteed (Checkbox)

Refers to the notional on the premium leg. The notional stays the same regardless of the number of defaults have occurred. Usually this is not checked, and the notional decreases at each default in the basket.

Amort Type

If "Senior" amortization is selected, the incurred recovery will be taken into account in processing credit events and pricing. Otherwise, only incurred loss will affect payments and notional. Please see the section of this document on senior amortization and incurred recovery for a full description.

5.5 Pricer Parameters: CDS Baskets

ACC_MODEL

Only used prior to Version 10.0. Replaced by **INTG_METHOD_CR**.

Specifies the Accrual Model used to estimate the expected value of accrued premium on Pay Accrual swaps. Values are:

ACC_MODEL = 1: Simple midpoint approximation (“J” method)

ACC_MODEL = 2: Integral approximation (“H” method)

For more information, please see the description of Pay Accrual models in the single-name CDS section.

BE_INCL_ACC

Modifies the calculation of BREAKEVEN_RATE to include past accrued interest or not. If True: the breakeven rate will be the premium rate that sets the NPV to zero, including past accrued premium (since previous coupon date) at the breakeven rate. If False: accrual will not be included, so the trade effectively starts accruing today (on the value date) -- This is the desired setting if one wants to price hypothetical new trades for hedging purposes.

The future cashflow payment dates are the same in either parameter setting and are taken from the trade at hand; only the effective start date of the first cashflow changes, as it is set to the value date for BE_INCL_ACC = False.

BUCKET_TYPE

For pricers that use the one-factor copula model (OFM). Specifies the type of buckets into which the tranche losses are collected during calculation. Precision is limited by the size of the bucket: larger buckets are less accurate but are faster to calculate. The two choices of BUCKET_TYPE are as follows.

"HW": (Hull-White) Simple method that buckets the portfolio losses as integer multiples of the smallest individual asset loss.

"A": (Andersen) Searches for an optimal bucket size to minimize bucketing errors, then uses approximately homogeneous sub-baskets to speed the calculation.

Usually, "A" will be the more accurate method, and "HW" the faster one.

CORRELATION

The value of a manual constant correlation used for every pair of assets. This overrides the market data correlation matrix. Note: To solve for constant correlation given a market data matrix, use the pricer measure IIMPLIED_CORRELATION.

MODEL_SPEED

Allows user to choose tradeoff of speed versus accuracy in certain numerical models, such as the Gaussian Copula model of the NthLossOFM pricer. In that pricer, a Model Speed of 1 is the slowest and most accurate, 2 is faster and of good general-purpose accuracy, and 3 is the fastest and most useful for sensitivity and jump to default analyses. In addition, two special purpose values, 64 and 128, have been added to check greater accuracy; these correspond to the number of factor values used in the Gaussian integration.

NUM_SCENARIOS

The number of scenarios to be run in computing the NPV if the Monte Carlo method is used. A value of 50,000 is suggested as providing reasonable speed and accuracy (0.4% standard deviation). A value of 500,000 provides an increase in accuracy (0.14% standard deviation) at the cost of an approximate tenfold increase in calculation time.

OBS_FREQ

(PricerCDSNthLossOFM) The observation frequency specifying the number of times per year that sampling is done to find the expected loss as a function of time. Interpolation is done between the sample points as needed.

PER_REF_NAME

If set to true, then the PV01_CREDIT is computed for each name in the basket by shifting its corresponding credit curve. To view the results, double-click on the PV01_CREDIT field in the trade window. This will bring up a window showing the sensitivity per name and the hedge ratio, i.e., the proportion of the total pv01 credit that is contributed by each name.

RECOVERY_MODEL

On PricerCDSNthLossOFM and PricerCDSIndexTrancheOFM; also a generator parameter on the Base Correlation surface generator. Controls the use of the Random Recovery extension of the Gaussian copula model. The choices on the pricers are **As_Corr_Surface, Fixed**, and **Random**. The default is As_Corr_Surface, which specifies that the pricer is to use the same model that was used to generate correlations in its correlation surface. But by selecting one of the other choices the user can force the choice of model regardless of the surface. This is useful for testing with manual input, for example. (See the Section on the Random Recovery Model for more details.)

RECOVERY_RATE

The recovery rate used in calculating credit default payments. The same rate is used no matter which asset defaults. If no rate is specified here, a rate is obtained from the probability curve or the recovery curve associated with each asset.

USE_STORED_SEED

For Monte Carlo evaluation. If set to true, the hard-coded initial value of the random number generator will be used, so that the same sequence of random numbers (for the given number of scenarios) is used on each evaluation. This eliminates random noise when comparing NPV values for different market data inputs.

Section 6. Monte Carlo Evaluation of CDS Basket Swaps

6.1 Procedure

Calypso evaluates CDS basket swaps using Monte Carlo evaluation. The approach follows that described by David X. Li, "On Default Correlation: A Copula Function Approach" (RiskMetrics Group, 2000).

The basic procedure is as follows. For the given basket of assets, a survival probability curve must be given for each asset, together with a matrix giving correlations between asset returns. One future scenario for default is drawn randomly from these probabilities; a scenario consists of a set of future default times, one time for each non-defaulted asset in the basket. For this scenario, the total payments over the trade lifetime can be determined as specified by the parameters of the trade. These payments are discounted to the present using zero curves. This process is repeated for a number of randomly selections of scenarios and the average over all the present values is taken to produce the NPV.

Repeated runs of the Monte Carlo pricing produces a distribution of NPVs. When the number of scenarios is 50,000 for one run, the NPV varies by about one half of one percent, while 500,000 scenarios produce an accuracy of about one tenth of one percent.

The essential part of the procedure is the production of default times with the appropriate joint probabilities of default. The method used in Calypso is as follows (please refer to Li's article for more information).

1. From the standard normal random (Gaussian) distribution, draw M independent values, where M is the number of assets (entities) in the basket.

2. Multiply the vector of M values by the Cholesky decomposition of the given correlation matrix. This produces a set of values x_i, i = 1,…,M representing a draw from a multivariate Gaussian distribution with the required correlations.

3. Convert the random variables x_i to default times t_i using the cumulative default probabilities $F_i(t)$. The transformation is:

$$t_i = F_i^{-1}(N(x_i))$$

Here N is the standard normal cumulative distribution function, and

$$F_i(T) = \Pr(t_i < T)$$

is the cumulative probability that a default has occurred by time T in the i^{th} asset. Note that in Calypso, the CurveProbability objects are survival curves, that is, they specify the survival probability S(t) that the entity has not defaulted by time T. Thus at time T = 0, the probability curve has S(0) = 1.00 (100%) for a non-defaulted entity. The curve then declines as time increases. Thus the probability of default is obtained from CurveProbability objects using:

$$F_i(t) = 1 - S_i(T)$$

4. Order the times to default t_i in chronological order (note *i* is the index of the asset in the basket, not the order of default). Using these, determine the first-to-default asset, second-to-default asset, etc., and use these to determine the cashflows of the contract under this scenario.

6.1.1 CDSNthLoss (CDO Tranche) Evaluation

A CDO tranche begins making payments when the losses from default reach an *attachment point* A and stops making payments when losses exceed a *detachment point* E. Premium is paid on the notional of the tranche, which initially is the difference E – A and which decreases as losses exceed A. The definition of the loss of an asset can depend on the contract, but typically it is the unrecovered portion of the asset notional. If R is the recovery rate of an asset and I it's notional amount, then the loss is (1-R)I.

Monte Carlo simulation of a tranche proceeds as described above for Nth defaults. For each scenario, the defaults are determined and the cumulative loss from the defaults is found for each payment period. The loss that is applicable to the tranche is then determined. If L(t) is the cumulative loss at the payment period ending at time t, then the loss applicable to the tranche is

$$L_{tranche}(t) = Min\,(E - A,\, Max\,(L_i - A, 0))$$

From these values the default payments and the premium payments can be computed for this scenario. Finally the average over all scenarios is taken.

For more details on the computation of net present value using the tranche loss, see the description below of the computation of NPV in the section on the one-factor model.

6.2 Implementation

This section describes the Calypso classes that are used in implementing the Monte Carlo valuation of CDS baskets.

`com.calypso.tk.pricer.RandomMultivariate`

RandomMultivariate is a general-use class providing the ability to draw random samples from a multivariate normal distribution with a given correlation matrix. It performs the first two steps of the procedure described above, drawing a set of independent standard normal random variables and multiply by the Cholesky decomposition matrix to obtain correlated samples. The class creates the Cholesky decomposition of the given correlation matrix (using matrix utility classes).

`com.calypso.tk.pricer.RandomMultivariateCreditDefault`

RandomMultivariateCreditDefault is a CDS-specific class that performs selections of credit default times drawn from a multivariate distribution of correlated probabilities of default. It subclasses RandomMultivariate, and performs steps 1 to 3 of the procedure described above. This object is set up and used by the Monte Carlo class. The following shows how to initialize and use the object:

```
//Instantiate the object.
       RandomMultivariateCreditDefault rm = new RandomMultivariateCreditDefault();
//Set the number of times needed to the number of assets in the basket
       rm.setNumberOfVariables(assets.size());
//Collect up probability curves in array, keeping order same as assets collection.
       PricerCDSNthDefaultInput anInput = (PricerCDSNthDefaultInput) pricerInput;
       CurveProbability[] arr = new CurveProbability[assets.size()];
       Hashtable cvDict = anInput.getProbabilityCurves();
       for (int j = 0; j < assets.size(); j++) {
        arr[j] = (CurveProbability) cvDict.get(assets.get(j)); }
       rm.setSurvivalCurves(arr);
//Set the correlation matrix.
//If none is set, the random generator will use the identity matrix.
       rm.setCorrelationMatrix(toMatrix(assets, anInput.getCorrelationMatrix()));
//Whenever a new set of random times is needed, do the following. The method
//drawNextDefaultTimes performs the draw and stores the result in the
//latestDrawTimes instance variable of the object.
       rm.drawNextDefaultTimes();
       defaultTimes = rm.getLatestDrawTimes();
```

`com.calypso.tk.pricer.MonteCarloValuation`

MonteCarloValuation is the base class providing instance variables of use in valuing trades using Monte Carlo. Instnace variables include numberOfScenarios, pricer, pricerInput, pricingEnv, trade, valDate, randomNumberGenerator, and useStoredRandomNormals (this last is discussed below).

`com.calypso.tk.pricer.MonteCarloCDSBasket`

MonteCarloValuationCDSBasket evaluates credit-default swaps on baskets of reference assets using Monte Carlo evaluation. It extends MonteCarloValuation.

The method to call for evaluation is calculateNPV(). All other methods and instance variables on the class support the NPV computation. Many of the instance variables hold cached values to speed up repetitive calculations. The following example shows how to use the object.

```
//Example of use:
//Create the object and set the instance variables:
  MonteCarloCDSBasket mc = new MonteCarloCDSBasket();
  mc.setTrade(trade);
  mc.setPricingEnv(env);
  mc.setValDate(env.getJDate(valDatetime));
  mc.setPricer(this);
  mc.setPricerInput(input);
  mc.setNumberOfScenarios(50000);
//Then use the object to compute net present value:
  mc.calculateNPV();
```

The calcualteNPV method sets up an instance of RandomMultivariateCreditDefault and repeatedly calls it to generate scenarios of times to default. Those scenarios are used to separately evaluate the premium leg and the credit default leg, the result summed and the average over all scenarios taken.

`com.calypso.tk.pricer.PricerCDSNthDefault`

PricerCDSNthDefault is the Pricer class Calypso provides for CDSNthDefault product. It extends PricerCreditDefaultSwap. To compute NPV this class instantiates a MonteCarloCDSBasket object; repeated calls to NPV provide delta and gamma values.

`com.calypso.tk.pricer.PricerCDSNthLoss`

PricerCDSNthLoss the Pricer class Calypso provides for CDSNthLoss product. It extends PricerCDSNthDefault. Since the handling of both Nth Loss and Nth Default is performed by the MonteCarloCDSBasket object, this pricer does not presently add any behavior to its superclass.

6.2.1 Storage of Random Numbers

In order to empirically compute sensitivities to changes in market data, repeated computations of NPV need to be performed. However, in a Monte Carlo method the NPV is only approximate and varies from run to run. The change in the NPV due to noise in the random draws can be larger than the sensitivities themselves. In order to eliminate this noise, the Monte Carlo classes in Calypso allow the ability to re-use the set of random numbers.

The simplest way to do this is to set the `useStoredSeed` parameter to `true`. A hard-coded value (the "seed") that initializes the random number generator will be used. As a result, the same sequence of random numbers will be used each time the Monte Carlo evaluation is run, even in different sessions.

Thus day-to-day changes in the value of the trade would be due only to the changes in market data, and not due to random noise. If one changes the number of scenarios, however, the length of the random number sequence will change, resulting in a different NPV.

If not using a stored seed one can still re-use the random numbers during one call to the price method of the pricer. The first time the method calculateNPV is called on an instance of MonteCarloCDSBasket is called upon to compute an NPV, the independent Gaussian random numbers generated for the calculation are stored in the MonteCarloCDSBasket object. More precisely, the object holds onto a random number generator, and instance of RandomMultivariateCreditDefault, which stores the generated collection.

The stored number are independent random Gaussians, and do not include transformation by the correlation matrix or the mapping to default times using probability curves. The matrix, curves and other market data can therefore be altered and the same random numbers used to determine the effect of that change without the noise of new random number generation.

Thus, repeated calls of `calculateNPV` to the same instance of MonteCarloCDSBasket, with the same market data, will result in the same NPV being returned. The Pricer object that uses the MonteCarloCDSBasket instance will compute interest-rate delta and gamma by shifting the discount curves and then calling `calculateNPV`; the storage of the variables ensures that only the effect of the curve shifting is seen in the NPV change.

The stored random numbers are thrown away when the MonteCarloCDSBasket object is discarded, which occurs when the Pricer has finished running its price method and computed all measures. There are situations where it may be desirable to store the random numbers for re-pricing at different times. To compute mark-to-market P/L, for example, one wants to compare today's NPV with yesterday's NPV, but in a Monte Carlo evaluation this number will have substantial noise if different random numbers were used in their scenarios. An improvement would be to store the numbers in a database and use them again each time NPV is computed. In this case, rather than generating the numbers in Step 1 of the procedure, they would be loaded from a file. This persistence is not yet available in the MonteCarloCDSBasket, but in future may be added. Possible persistence for P/L consistency.

If one does not want to use the stored random numbers feature, then when initializing the MonteCarloCDSBasket object set `useStoredRandomNormals` to `false`.

Section 7. General Tests and Performance: Monte Carlo

7.1 Performance

Basket sizes that are typically traded usually range between 5 and 10 assets. The following performance tests were performed for a first-to-default trade with five years to maturity. The processor was a Pentium 4 and there were thirty runs performed for each row of the table.

In the results, note the standard deviation of the NPVs decreases as the square root of the number of scenarios, as expected. The average time per NPV computation increases approximately linearly with basket size and number of scenarios.

Basket size	Num scenarios	Average time/run (seconds)	Observed st dev
3	10000	0.10	0.76%
3	50000	0.48	0.27%
3	500000	4.77	0.09%
10	10000	0.91	0.56%
10	50000	4.58	0.25%
10	500000	45.66	0.08%

7.2 Example Values

The following table gives the breakeven premium rates for various basket swaps in terms of the underlying single-name default swaps. In each case it is supposed that the assets in the basket have the same probability curve. A recovery rate of 30% is used. The correlation given means that all of the asset pairs have that correlation.

To use this for testing, find the breakeven premium rate for a single-name five-year CDS trade. Create a basket of five assets that use the same probability curve as this CDS. Multiply the single-name CDS premium rate by the factor in the table to find the breakeven premium rate for a given basket CDS.

Reference: *Credit*, A. Arvanitis & J. Gregory (Risk Waters Group, London, 2001), Ch. 5

Multiplicative factor for premium rate of 5-asset basket.

Product	0% correlation	25% correlation	50% correlation
First to default	5	4.36	3.57
Second to default	.57	.88	1.12
Third to default	.04	.19	.42

Fourth to default	0	.03	.14
Fifth to default	0	0	.03
First two to default	2.59	2.49	2.28
Last three to default	0	.02	.09

7.3 Relationships Among CDS Basket Values

Certain relationships among different CDS basket products are expected based on arbitrage arguments. The dependence of their valuation on the correlations in the basket can also be understood. These plausibly relationships can be used as general tests of the Calypso implementation.

To test these relationships in Calypso one should set up the appropriate trade and then solve for the Breakeven premium rate. When relationships among premiums are discussed below, the Breakeven premium rate is meant.

Reference: *Credit*, A. Arvanitis & J. Gregory (Risk Waters Group, London, 2001), Ch. 5

First to default

For a first-to-default basket swap, the trader has the choice between buying a basket CDS or buying a single-name CDS on each of the individual assets in the basket. If the assets are completely uncorrelated one expects there to be no difference between these trades, and so the premium to be paid for protection of the basket should be the sum of the premiums on the single-name swaps. For an increase in correlation of the assets, the likelihood that more than one single-name swap will default increases, representing protection on more than one asset; hence the premiums for the sum of the single-names should be more than that of the basket, which provides protection on only the first of the defaults. Thus one expects the premium of the basket to become less than the sum of the individual CDS premiums as the correlation increases. To summarize:

	Increase correlation	**Zero correlation**
1st to default	Less than sum of individual premiums, greater than the largest one	Equals sum of individual premiums

Nth to default

For any given set of correlations, a second-to-default swap is less likely to encounter a credit default event than a first-to-default swap, as two defaults are less likely than one. So protection should cost less for the second-to-default swap, and in general the Nth to default swap should have a lower premium the larger N becomes.

For a given N, increasing the asset correlations will increase the probability that N defaults actually occur if one occurs. Thus more protection is needed; the premium will increase as correlation increases.

In summary:

	Increase correlation	Increase N
Nth to default	Increase premium	Decrease premium

First two to default

A basket default swap that pays on both of the first two defaults ("1-to-2" defaults) can be compared to a combination of a first-to-default ("1-to-1") and a second-to-default ("2-to-2"). Suppose the notional amount of each protected asset is A. The buyer of protection on the first-two-to-default pays a premium rate R(1,2) on a notional that starts at 2A (since two assets are protected) and, after one default, decreases to A. Suppose the first-to-default premium rate is R(1,1) and the second-to-default rate is R(2,2). Together the protection payments for the default events are the same as a basket swap, so the total premium amount paid must be the same as the basket. Each of these latter default swaps pays on a notional of A; suppose the first default happens at time T1 and the second at time D later. Then setting the premiums of the basket equal to the total of the individual swaps gives

$$2R(1,2)T1 + R(1,2)D = R(1,1)T1 + R(2,2)(T1+D)$$

Now on the left

$$2R(1,2)T1 < 2R(1,2)T1 + R(1,2)D$$

and on the right

$$R(1,1)T1 + R(2,2)(T1+D) < (R(1,1)+ R(2,2))T1$$

so

$$2R(1,2) < R(1,1) + R(2,2).$$

Also, if no default occurred the 1-to-2 swap should have had to pay out total premiums greater than the single-protection 1-to-1 swap, so it must be that

$$R(1,1) < 2R(1,2).$$

These give bounds R(1,2) premium against the individual swaps:

First two to default	$R(1,1) < 2R(1,2) < R(1,1) + R(2,2)$

This can be used to check the 1-to-2 swap premium.

One also has:

Last m to default	Increase premium (but premium always small)

7.4 Analytic Examples

For cases of simple correlations and probability curves, analytic values for the credit default payments on basket default swaps can be found.

To define a probability curve, use a constant hazard *rate h*, which is related to the probability of survival at time t by

$$S(t) = e^{-ht}$$

Calypso probability curves are survival curves, so this expression can be used to manually create probability curves for a given hazard rate. (A typical value for good quality assets is h = 0.10.) Then for simple correlations the value of the protection leg can be found by integrating the discounted payments times the probability of default in each small time interval. This gives the following results:

Product type: Nth to default basket

Number of assets: n

Time to default (years): T

Constant zero rate: z

Constant hazard rate: h

For completely uncorrelated assets (0% correlation), the protection leg NPV is:

$$\frac{nh}{z+nh}(1-e^{-T(z+nh)})$$

For completely correlated assets (100% correlation), the protection leg NPV is:

$$\frac{h}{z+h}(1-e^{-T(z+h)})$$

<u>Reference</u>: David X. Li, "On Default Correlation: A Copula Function Approach" (RiskMetrics Group, 2000).

Section 8. One-Factor Model

Factor models provide a much more rapid means of evaluating CDS Baskets than can be obtained using Monte Carlo. These use the same copula concept that underlies Monte Carlo evaluation, but employ a reduction of the dimensionality of the correlation matrix by means of a factor representation.

The greatest simplification is obtained by using a "flat" correlation matrix in which all off-diagonal elements are the same value, representing in principal an "average pair-wise correlation." The single average correlation can be represented by a *one-factor model* of the default probabilities. The CDS marketplace has been tending toward the use of a Gaussian copula one-factor model as a standard.

The one-factor model is accessed in Calypso using the `PricerCDSNthLossOFM` and `PricerCDSNthDefaultOFM` and is implemented with a set of classes that allow extension to non-normal probability distributions and to multi-factor models.

The classes implement these steps to calculate the NPV of a CDO tranche:

1. Compute the loss distribution for the entire basket
2. Use the loss distribution to compute the average loss for the CDO tranche.
3. Use the average loss of the tranche to find the NPV.

Steps 2 and 3 are common to all models, and Calypso provides classes to compute these independent of the model used to define the loss distribution. A class to compute the one-factor model is provided to implement Step 1, independently of the NPV computation. Thus new models can be added to Calypso by overriding the model of Step 1. This chapter describes the mathematics and classes in detail.

An extension of the Gaussian copula is the **Random Recovery model**, which was introduced to handle the extreme pricing of index tranches in the 2008 credit crisis. For more information please refer to the "Random Recovery Model" section later in this document and the Calypso white paper, "Random Recovery Models for CDO Tranche Pricing".

References:

Leif Andersen, Jakob Sidenius, Susante Basu, All your hedges in one basket, RISK, November 2003

John Hull, Alan White, Valuation of a CDO and an nth to Default CDS without Monte Carlo

Simulation, Working Paper, September 2004

Philipp Schönbucher, *Credit Derivatives Pricing Models* (Wiley 2003)

Calypso, Random Recovery Models for CDO Tranche Pricing (Calypso 2008)

8.1 One-Factor Model Loss Distribution

8.1.1 Correlation Modeling

The one-factor model employs a simple copula, which is based on the following observation. Let X1, X2, and M be three *independent* random variables. Let the mean of each be zero and the standard deviation equal to one, but otherwise their probability distributions need not be the same. Then the two random variables

$$Y_1 = \sqrt{c}\,M + \sqrt{1-c}\,X_1\ ,$$
$$Y_2 = \sqrt{c}\,M + \sqrt{1-c}\,X_2$$

each have mean zero, standard deviation of one, and are correlated with correlation coefficient *c*.

Conversely, given a pair of correlated variables (the Y's) one can create a triplet of uncorrelated variables (M and X's).

For CDS Basket trades, define:

p_k (t): probability that the kth asset has defaulted by time t;

k = 1,…, K_B;

K_B: number of assets in the basket that have not yet defaulted.

Associate with the kth asset a random variable Y_k which is presumed to govern default; for example, Y_k might represent the asset value , and default occurs if this value falls below a certain threshold. Then a value of y_k can be associated with a time to default *t* by specifying that the probability that $Y_k < y_k$ is equal to the probability that default has occurred by time *t*:

$$p_k(t) = \Pr(Y_k < y_k)$$

The Y_k's are correlated, so introduce uncorrelated variables X_k by means of a common factor M:

$$Y_k = \sqrt{c}\,M + \sqrt{1-c}\,X_k$$

Then if M has value m with probability Pr(m), one can write:

$$\begin{aligned} p_k(t) &= \Pr(Y_k < y_k) \\ &= \sum_m \Pr(\sqrt{c}\,m + \sqrt{1-c}\,X_k < y_k)\Pr(m) \\ &= \sum_m \Pr(X_k < \frac{y_k - \sqrt{c}\ m}{\sqrt{1-c\,m}})\Pr(m) \end{aligned}$$

For any given value m of M, define

p_k(t | m): conditional probability that the kth asset has defaulted by time t when M takes on the value m, so that

$$p_k(t) = \sum_m p_k(t|\,m)\Pr(m)$$

Then we have

$$p_k(t|\,m) = \Pr(X_k < \frac{y_k - \sqrt{c}\ m}{\sqrt{1-c\,m}})$$

Now it is necessary to choose a form for the distributions of M, the X's, and the Y's. Following market practice, assume each has a Gaussian distribution. Then

$$p_k(t) = N(y_k)$$

with N(y_k) the cumulative standard normal distribution function; and so

$$p_k(t|m) = N(\frac{N^{-1}(p_k(t)) - \sqrt{c}\ m}{\sqrt{1-c\ m}})$$

The probabilities conditional on M are uncorrelated, depending on the distribution of the independent X_k. Using these it is a simple matter to compute the probability of default of any set of assets, as described in the next section.

8.1.2 Loss Distribution Calculation

The cumulative loss distribution is the probability that the losses of the entire basket of assets reach a given value at any given time. Define:

P(L, t): probability that basket has loss of L at time t

Denote by L_k the loss of an individual asset, which is the part of the asset's principal that is not recovered in the event of default:

$L_k = I_k(1 - R_k)$

with

I_k: Principal of the kth asset

R_k: Anticipated recovery rate of the kth asset

Since not every value of L can be obtained by summing the L_k, define a loss amount dL and solve for the probability that the total loss is between L and L + dL.

The calculation of the loss distribution for a given L and t follows these steps:

1. Find all sets of assets whose summed loss falls between *L* and *L* + *dL*. If S_i is one such set, given by a list of the k-indexes of the defaulting assets, then

$$L <= \sum_{k \in S_i} L_k < L + dL$$

2. Select a value of the factor *m*.

3. Conditional on m, compute the probability of each set in Step 1, that is, the probability that the assets in the set default and the other assets in the basket survive. Because the default probabilities are independent conditional on m, the probability of a set S_i occurring is

$$\Pr(S_i \mid m) = \prod_{k \in S_i} p_k(t \mid m) \prod_{k \notin S_i} (1 - p_k(t \mid m))$$

4. Sum over all asset sets found in Step 1 to give the loss probability conditional on m:

$$P(L,t \mid m) = \sum_i \Pr(S_i \mid m)$$

5. Repeat Steps 2 through 4 for all values of *m*.

6. Finally sum the results for all *m*, weighted by the probability of *m* occurring:

$$P(L,t) = \sum_m P(L,t \mid m) \Pr(m)$$

For a Gaussian copula, Pr(m) is the normal density function times the distance between successive values of m (to approximate the integral over continuous m).

The first step is the most computationally time consuming: finding all asset losses that sum to a given loss. Andersen et al. give an efficient procedure for organizing this calculation of P(L,t), a variation of which is found in H & W (see the References). This has been implemented in Calypso in `tk.model.credit.OneFactorLossDistribution`.

8.2 Calculating NPV from the Loss Distribution

Given the loss distribution P(L, t), the net present value of a CDO can be computed. Let

A: Attachment point of the tranche

E: Detachment (exhaustion) point of the tranche

These are expressed as monetary amounts (not percentages), and define the current tranche that determines future payouts – that is, past losses from the initial basket modify the initial attachment/detachment points to provide the current A and E points.

The NPV of this tranche is computed as follows.

1. Find the *expected tranche loss* for each sample time *t*.

$$L_{tranche}(t) = \sum_i P(L_i, t)\ Min\,(E - A, Max\,(L_i - A, 0))$$

Here, L_i are representative losses for each loss bucket. A simple procedure that is exactly correct if all asset losses are equal is to let L_i be the lower boundary of each loss bucket; for unequal losses, a smaller error is found by using the average loss that occurred in each bucket.

2. Find the expected payments from defaults in each sample period. These are equal to the change in expected tranche loss in that period. Discount these payments to the present with discount factor *D(t)*. This gives the *credit leg NPV*:

$$NPV_{credit} = b\sum_t D(t)\ (L_{tranche}(t) - L_{tranche}(t-1))$$

Here, *b* is the sign of the payments, equal to +1 if buying credit protection (receiving default payments), and -1 if selling credit protection.

3. Find the expected premium payments for each sample period. The notional on which premium is paid is determined by the tranche loss. The expected notional at time *t* is the size of the tranche, *E – A*, minus the expected tranche loss, with the practical limitation that the notional cannot be greater than the total basket notional of the non-defaulted assets:

$$N_{tranche}(t) = Min\,(E - A - L_{tranche}(t), UndefaultedBasketNotional)$$

The need to take the minimum is the result of defining losses as the unrecovered part of the notional. Consider a "tranche" of 0 to 100%. If asset of notional I defaults, the tranche notional decreases by (1-R)I, while the entire basket notional decreases by I, making the tranche notional formally greater than the basket notional. But protection should only be paid for on no more than the remaining basket notional.

Contracts can vary in their definition of the notional, depending on whether no accrual, full accrual, or a riskless payment is made. For premium payments that pay no accrual, the applicable notional is only that which occurs at the end of the period. Then for premium rate (credit spread) *r* and period length *a(t),*

$$NPV_{premium} = -b\sum_t D(t)\, N_{tranche}(t) r a(t)$$

Similar formulas apply for other types of accrual.

4. Finally, the net present value of the trade is the sum:

$$NPV = NPV_{credit} + NPV_{premium}\,.$$

8.3 Non-Homogeneous Small Basket Nth Default Model

When using PricerCDSNthDefaultOFM, a small basket model will be automatically employed if the number of assets in the basket is ten or less and the recovery rates are not all equal. This model will calculate the probability of every combination of defaults and survivals of the assets on each sample date and use this information in finding the probability and value of an m-to-n default.

The method implements the non-homogeneous method of J.-P. Laurent & J. Gregory, "Basket Default Swaps, CDO's and Factor Copulas" (2003), Section 4.4.

The Model

This section summarizes the argument of Laurent and Gregory.

Consider the problem of valuing the protection payout for the nth default of a basket with M assets. Under the Gaussian copula, the survival probabilities for each asset are independent when conditioned on the copula factor. Let pk(t) denote this conditional survival probability for the kth asset, so the pk are all independent; the value of the copula factor is suppressed in this notation. At a given time, the probability of any combination of survivals and defaults (for this factor) can be readily computed. There will be 2M such combinations.

For example, in a basket of 5 assets, denoted by {11001} the combination in which the first and second asset have survived to t, the third and fourth have defaulted, and the fifth has survived (1 denotes survival, 0 denotes default). The probability of this combination is:

$$P(\{11001\}) = \; p_1(t)p_2(t)(1 - p_3(t))(1 - p_4(t))p_5(t)$$

This basket has 32 such combinations.

Having computed these probabilities at each sample time, the probability that the n^{th} default occurs between successive times t_i and t_{i+1} is found by taking the difference

Probability of n defaults at time t_{i+1} - Probability of n-1 defaults at time t_i

where the probability of n defaults is just the sum of the probabilities of all combinations which have n defaults.

For a homogeneous basket, where all assets have the same expected recovery rate, this calculation is sufficient to compute the expectation of the loss payouts for an nth default: simply multiply the probability of an nth default between two sample points by the loss payout 1-R, apply a discount factor for that period, and sum over all periods. However, for a nonhomogeneous basket one must know the probability that each asset is the nth default, and weight its payout 1-Rk accordingly.

To perform the nonhomogeneous calculation, select each asset and find the probability it will default between sample time t and a short time later, t+dt. This produces an approximation of the hazard rate for default at time t. For the kth asset,

$$\frac{dp_k}{dt} \approx \frac{p_k(t+dt) - p_k(t)}{dt}$$

Consider the basket with the kth asset removed; find all combinations of this reduced basket which have n-1 defaults. Denote the summed probability of these combinations as P(n-1|k,t), that is, the probability at time t of n-1 defaults on the basket with k removed. Then the probability that the kth asset will be the nth default of the full basket is

$$-P(n-1|k,t)\frac{dp_k}{dt}$$

and the expected value of the nth-default payment at time t is this probability times the notional of the asset times 1-Rk, where Rk is the recovery rate of the kth asset at this time. Repeating this procedure for each asset produces the expected payout for an nth default at time t, and multiplying by the discount factor D(t) produces the present value:

$$PVExpectedPmt(t) = -D(t)\sum_{k=1}^{M} P(n-1|k,t)\frac{dp_k}{dt}(1-R_k)N$$

Note that it is assumed all assets have the same notional amount N, which is typical for an nth-default (if the notionals were not all equal, additional provisions would be needed in the contract to remove the ambiguity in the payout amount).

Repeating this at each sample time produces the instantaneous payout at that time. Then the expected value of the payments over the lifetime of the contract is found by integrating over all time, using interpolation to find times between the sample times. Currently Calypso employs linear interpolation to perform the integration. However, approximation error is best reduced by increasing the number of sample times. Daily sampling achieves the smallest error, which in Calypso is achieved by setting OBS_FREQ to the number of days per year (it can just be set to 400 as the frequency is rounded to an integer number of days).

This completes the calculation for one value of the copula factor. The numerical integration over factors is then performed: the preceding calculation is repeated for each factor in the grid of the numerical integral, weighted by the appropriate Gaussian factor and summed.

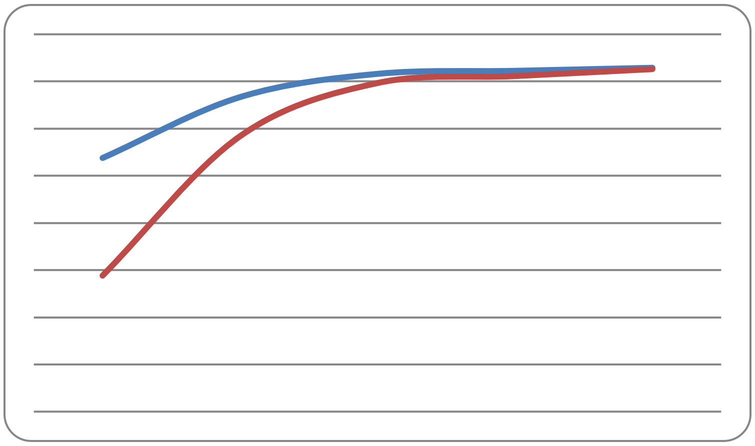
$$NPV = \int_{-\infty}^{\infty} f(v)dv \int_{0}^{T} PVExpectedPmt(t;v)\, dt$$

where the dependence of the conditional probabilities on the factor value v has been made explicit.

For an n-to-m default, instead of simply solving for the value of the nth default at a sampling time one varies the number of defaults between n and m, repeating the calculation for each and summing.

As stressed above, the sampling frequency plays a role in the accuracy of the calculation. As an example, consider the homogeneous case where all recoveries are the same, and compute it in two different ways. In the first method one uses the simple homogeneous calculation, while in the second case one uses the nonhomogeneous method of approximating the hazard rate of each asset to find the instantaneous payment, then integrating. The following table compares the NPV of the protection payment for a small basket first-to-default when computed at various sampling frequencies:

Obs freq	NPV Homog	NPV Nonhomg
4	1,556,890.10	1,544,421.78
12	1,563,345.33	1,559,197.29
52	1,565,807.67	1,564,836.43
100	1,566,129.68	1,565,574.93
400	1,566,451.41	1,566,312.72



The methods converge as sampling frequency increases.

8.4 Recommended Parameter Settings: Speed Versus Accuracy

The speed of the calculation can be varied by the user by setting parameters on the model. As always with numerical methods, there is a tradeoff of speed versus accuracy. The following table shows recommendations of parameter settings for speed/accuracy choices.

Recommended Parameter Settings

Speed versus Accuracy	MODEL_SPEED	OBS_FREQ	BUCKET_TYPE	INTG_METHOD_CR	Comments
Default	(none set)	(none set)	A	LINEAR_SINGLE	Defaults to Model Speed = 1 and an Obs Freq equal to the premium payment frequency (e.g., 4 times per year).
Highest Accuracy	1 (or 64 or 128)	4	A	SIMPSON or EXACT	Slowest speed, highest model

Speed versus Accuracy	MODEL_SPEED	OBS_FREQ	BUCKET_TYPE	INTG_METHOD_CR	Comments
					accuracy. Can be too slow for sensitivity analysis. Best for small portfolios.
Balanced Recommendation	2	1	A	SIMPSON	Good speed and accuracy. Recommended all-purpose settings for indexes and other large portfolios.
Highest Speed	3	1 or 0.5	HW	LINEAR_SINGLE	Highest speed. Can be best choice for sensitivity and Jump to Default analysis where NPV differences are computed, but may not be acceptable for pure NPV.

OBS_FREQ (Observation Frequency) specifies the number of times per year that sampling is done to find the expected loss as a function of time. Between sample dates, linear interpolation is done to complete the expected loss function. (Spline interpolation is not used as it is possible to arrive at unrealistic results, such as negative loss over a period.) For large portfolios, a frequency of one sample per year is typically found to give results very close to those of higher frequencies.

MODEL_SPEED summarizes choices for other parameters in the numerical approximation of the loss distribution. Values of MODEL_SPEED are integers that are mapped to internal parameter values, with increasing values corresponding to faster speeds -- a MODEL_SPEED value of 1 represents the slowest choice, 2 the next faster, 3 the fastest.

In calculating the loss distribution, one approximation involves the number of values of the factor (*m*) over which the integration is performed, and the method of integration. Calypso employs Gaussian quadrature with the number of factors corresponding to the number of weights used in the quadrature. Changing MODEL_SPEED will change this number of factors, with fewer factors used at higher MODEL_SPEED values.

A second source of approximation involves the choice of loss buckets. This is the dL of Step 1. The most accurate case occurs where all assets have the same estimated losses. Then bucketing can be performed without error by choosing the bucket size dL to be equal to the loss of one asset. An example of this case is where all the assets in the basket have the same notional amount and all the assets have the same estimated recovery rate. For index products, the former is true but the latter -- equal recovery rates -- usually is not. So in practice there is always an approximation involved.

By choosing very small loss buckets, and a very large sampling of the factor, the error from the numerical approximation can be made small, but at the cost of greatly increased processing time. Calypso's implementation is intended to strike a balance between speed and accuracy. This implementation produces an improvement in speed of a factor of 100 over a Monte Carlo calculation of similar accuracy.

For baskets in which there are *very large differences in asset notionals or recovery rates* (ratios of 5 or 10), Monte Carlo is recommended as the more accurate method, or else the one-factor code can be modified to decrease the bucket size. Currently MODEL_SPEED will not affect the size of the loss buckets.

For the basic OFM model, at MODEL_SPEED controls the number of Gauss-Legendre points used in the numerical integration over the conditioning factor. The fewer points used, the faster the valuation.

Model Speed	Num points
0 or 1	32
2	16
3	7
64	64
128	128

The default is speed of 0 (equal to 32 integration points). Using 32 points is usually quite an accurate evaluation for the functions in the OFM model, but for verification purposes one can select also 64 or 128 points. The 16 points may be quite sufficient, and even 7 points usually gives good results.

When doing Jump to Default and PV01 Credit per name calculations, we add another shortcut to speed up evaluation. A speed of 0 or 1 won’t include this shortcut, so will have exactly the same valuation as the default valuation of the OFM. But using one of the higher speeds will also override the samples per year used in the loss computation. This is the same quantity controlled by the OBS_FREQ pricing parameter. Thus when doing Jump to Default or per-name sensitivities, we use the above integration points and also:

Model Speed	Jump-to-Default Analysis Samples per year
0 or 1	User’s OBS_FREQ (default 4, quarterly))
2	1.0 (annual)
3	0.5 (once every two years)
>3	0.25 (once every four years)

Note the user’s OBS_FREQ is overridden only if MODEL_SPEED is 2 or higher. In addition, for the case of trade maturity too small for the requested sampling frequency (less than one sample to maturity) the sampling frequency is increased.

8.5 Calypso Classes: How to Modify or Add Credit Models

While a new credit model can be added by writing a new pricer from beginning to end, the Calypso classes provide reusable functionality that can shorten the time of development. In particular, a new model that computes only a loss distribution, and nothing else, can be added by the user, and it will be picked up by the other Calypso classes for computation of new present value, sensitivities, and all other pricer measures, without further work on the developer's part.

The following table shows how the classes divide responsibilities for the calculations.

Package	Class	Responsibility
tk.pricer	PricerCDSNthDefault, PricerCDSNthLoss	Defines pricer measures in terms of NPV, sets up market data, calls credit model
tk.pricer	PricerCDSNthLossOFM PricerCDSNthDefaultOFM	Calls the one-factor model implemented in OneFactorLossDistribution. All other behavior is inherited from PricerCDSNthLoss or PricerCDSNthDefault.
tk.model.credit	CDSModelInterface	Interface required to be implemented by credit models for use by the pricer
tk.model.credit	CDSBasketLossDistributionModel	Computes NPV from a loss distribution. Implements CDSModelInterface.
tk.model.credit	OneFactorLossDistribution	Computation of the loss distribution for the one-factor Gaussian copula model. A subclass of CDSBasketLossDistributionModel.

To add a new loss distribution calculation – for example, a Student-t copula model – the developer can do the following:

1. Write a subclass of CDSBasketLossDistributionModel that computes the loss distribution given a set of assets and the market data. It should implement the method computeLossDistribution and/or computeExpectedTrancheLoss. Note: If the new model is a one factor model, the developer can subclass OneFactorLossDistribution and reuse many of the methods.

2. Create an empty Pricer that is a subclass of PricerCDSNthLoss. The name of the pricer should indicate the type of model used. (Example: PricerCDSNthLossOFM makes use of the Gaussian one-factor model.) If additional market data is needed for the model, this pricer should add a getInput method to set the data on the PricerInput (this may require creating a new subclass of PricerCDSNthDefaultInput in order to handle the extra data).

3. Add a createModel method to the new pricer that creates an instance of the model class defined in step 1. Here is the typical form:

```
public NPVModelInterface createModel(Trade trade,
          JDatetime valDatetime,
          PricingEnv env,
          PricerCDSNthDefaultInput input) throws PricerException {
      OneFactorLossDistribution ofm;
```

```
        ofm = new OneFactorLossDistribution();
        ofm.setTrade(trade);
        ofm.setPricingEnv(env);
        ofm.setValDate(env.getJDate(valDatetime));
        ofm.setPricer(this);
        ofm.setPricerInput(input);
        return ofm;
}
```

The new pricer will inherit the methods from PricerCDSNthDefault or PricerCDSNthLoss and at once be able to compute the pricer measures using the new calculation of the loss distribution.

Section 9. CDS Index and Index Tranche Pricing

A trade in a CDS Index – a spread on a published basket of reference entities – can for the most part be valued as any credit default basket swap. However, there are some differences arising from the conventions of the CDS Index market.

1. A CDS Index pays coupons at a premium rate (spread) that is fixed at inception. The market will quote the index at a current premium rate, and when a trade is made the present value of the difference between the market quote and the coupon rate is paid as an upfront fee. The cash settlement of index trades provides an alternative to valuation derived from probability curves.
2. The liquid market in index tranche products provides a means to imply correlations that can be used to price off-the-run indexes, aged trades and non-index basket swaps. This allows the concept of a *base correlation surface* generated from market prices and used in pricing these trades.

In this chapter the particulars of how Calypso values CDS index and CDS index tranche are described. In the subsequent chapters can be found a discussion of the generation of base correlation surfaces and of pricing non-index "bespoke" trades using index information.

Calypso provides functionality to price an index trade in two ways: either from an index quote, or from probability curves (and correlation surfaces if an index tranche). The choice is determined by the setting of the pricing parameter NPV_FROM_QUOTE. If the parameter is set to False, Calypso will convert the index trade into its equivalent CDSNthLoss trade and price it accordingly.

A CDS Index trade involves protection on the entire index portfolio. Thus it is equivalent to a CDO (CDSNthLoss) with an attachment point of 0% and a detachment of 100%. A CDS Index has physical settlement: when a default occurs, the party receiving the coupons pays the principal of the defaulted asset, and the other party delivers the same face value of that asset. The notional on which the coupons is paid reduces by that principal value.

Remark: The reduction by whole principle amount may require explanation. If an asset defaults, the notional of a 100% tranche on which premium is paid must decrease by the entire notional amount of that asset, rather than by the usual "1 minus recovery rate" loss amount for a typical tranche paydown. For example, suppose the portfolio has a notional principal of $100,000,000 in reference assets and one asset with principal $1,000,000 defaults with recovery rate 40%. A CDO tranche of 0% to 7% , say, would decrease its notional by 1,000,000 * (1 - .40) = $600,000. But if a 100% tranche decreased notional by the amount the remaining notional of the basket would be 99,400,000, which is larger than the 99,000,000 in assets that actually remain. Protection cannot be paid on more assets than in the basket, so after each default the basket notional is reduced by the entire notional amount, not just the loss amount.

References:

"iBoxx CDX.NA.IG: The New US Credit Default Swap Benchmark Index" (iBoxx, October 2003)

"Credit Derivatives Special - DJ iTraxx" (HVB, Oct 2004)

9.1 Valuation from Probability Curves (0-100% Tranche)

Whole-basket trades do not depend on asset correlations, as so there is a fast calculation of the net present value which does not use either Monte Carlo or a one-factor model. The value of the CDS index is the same as that of the credit default swaps on the individual reference assets of which it is composed. This calculation need not be done explicitly, as it is faster to find the loss distribution of the basket, as follows.

If the tranche covers the entire basket,, the equation for the tranche loss in terms of loss buckets L_i, described previously, reduces to:

$$L_{tranche,100\%}(t) = \sum_i P(L_i, t)\ L_i$$

The expected tranche loss is thus the same as the mean of the entire loss distribution. But this mean is independent of the correlation (as can be shown in general). A tranche that is narrower than 0 to 100% depends not only on the mean of the distribution but also on its shape, which determines the amount of the distribution that falls between the attachment and the detachment points. This shape is dependent on correlation, while the mean is not, which is why tranches less than 100% in width have a correlation dependence not found in whole baskets.

The expected tranche loss can therefore be computed as if the assets had zero correlation, by simply summing the expected loss L_k of each asset:

$$L_{tranche,100\%}(t) = \sum_k p(k, t)\ L_k$$

From this loss distribution the NPV of the credit and premium legs are found, as described in the previous chapter.

Remark: To see why the mean of the distribution is independent of the correlations, consider a two-asset basket. Let the assets have potential losses L_A and L_B. Let A denote default of the first asset at maturity of the trade, B default of the second, and A^c, B^c the complementary events, i.e., survival. The expected loss at maturity is found by summing over the probabilities of all combinations of events times their associated losses:

$$\overline{L} = P(A \cap B^c)L_A + P(A^c \cap B)L_B + P(A \cap B)(L_A + L_B) + P(A^c \cap B^c)(0)$$

This looks dependent on the correlations because of the joint probabilities, as

$$P(A \cap B) = P(A)P(B) + \rho(A,B)\sqrt{P(A)P(B)P(A^c)P(B^c)}$$

However, because

$$P(A \cap B^c) = P(A) - P(A \cap B),$$

$$P(A^c \cap B) = P(B) - P(A \cap B)$$

the sum over all possibilities causes the dependence on joint probabilities to disappear. By adding more assets one by one it can be seen inductively that the result holds for baskets of any size.

9.2 Valuation from Clean Price Quote (Price-Based Index)

A CDS Index can be quoted in the market in one of two ways:

1. in terms of a bond-like clean price, near 100 at inception, called a price-based index
2. in terms of a breakeven spread, following a particular convention for converting an upfront fee into the spread

In this section, the price-based index is described. This is the simplest method.

If quoted as a clean price, the NPV of a CDS Index is given by

$$NPV = wN(1 - p) + wA$$

where

N: Notional of trade (a positive number)

A: Accrued premium of current period (a positive number)

p: Clean price per 1.0 of notional (usually a number near 1.0)

w = +1 if paying the premium, 1 if receiving the premium

The NPV is zero if the index is priced at par, and can be positive or negative as price fluctuates.

This sign of the NPV requires special note. First, a common terminology is to use “buy” and “sell” of the index as though one were buying or selling the underlying bond. Thus:

“Buy the Index” = receive premium, sell protection

“Sell the Index” = pay premium, buy protection

Second, the sign in front of the clean price *p* is determined by the requirement that the index price acts as a bond price, moving inversely to the spread.

One sees the formula obeys this requirement. For the Buyer of the Index, receiving premium, the buyer’s NPV from the formula goes up as *p* increases. The only way the premium receiver’s NPV can increase is if the CDS spread decreases (so the receiver is being paid more than the market value of the coupons). So indeed the spread is moving opposite to the price.

9.3 Valuation from Index Spread Quote (CDSW Pricing)

Reference: "Credit Derivatives: A Primer" (JP Morgan, Jan 2005)

When indexes are traded, the market often computes the upfront fee from a quoted spread using the JP Morgan method, available on the Bloomberg CDSW screen (“Method J”). This provides a formal translation between a fee and a spread similar to bond price-yield conventions, but it does not imply that a new swap can be entered into at that spread.

Calypso provides this calculation of NPV in its CDS index pricers when the parameter NPV_FROM_QUOTE is set to True. It is also available in the single-name CreditDefaultSwap pricer when the user enters a value for the BE_RATE on the pricing parameter tab; and as the default method in pricer CreditDefaultSwapQuote.

Description of Method

The standard algorithm for pricing a credit default swap from a probability curve is employed, with these specifications when pricing from a quoted spread:

- The value of the future termination payments is the same as that of a hypothetical CDS starting on the settlement date, with the same cashflow dates as the given trade, whose breakeven premium rate is the quoted spread.
- The probability curve is a flat-spread curve, created by taking the quoted spread as the breakeven spread for credit default swaps of all tenors, starting on the settlement date. (Please see the next section for curve creation method.)
- The recovery rate is defaulted to 40%. (This can be changed for CDS Indexes.)
- Accrued premium is handled using Morgan approximations (see below).

The NPV and the Upfront Fee are not identical: they differ by the dates on which they are computed, as well as a sign. Dates are defined as follows:

Value Date: The date of the market data, including the discount curve and the index quote.

Settlement Date: The first day of protection that is to be included in trade valuation. If pricing a trade on an aged swap, this is the same as the date on which ownership of the swap changes hands. The seller of protection will receive the premium flows that occur strictly after the Start Date.

Fee Date ("Cash Paid On" Date): The day the Upfront Fee is actually due.

Start Date (Effective Date): The first day of protection of the swap. This will be in the past if an aged swap.

Spot Date: The Start Date of a swap for which a given index quote is made, usually Value Date plus one business day.

The Upfront Fee is defined as follows:

- If an aged swap is changing hands, the Upfront Fee generated from the quote is the negative of the NPV for the Spot Date. Cashflows that occur on the Spot Date will not be included in the Upfront Fee calculation.
- If a forward-starting swap, the Upfront Fee is the negative of the NPV for the forward Start Date.

In all cases, the market data used is as of the Value Date.

Note that, in contrast with the Upfront Fee, the NPV Pricer Measure is computed on the Value Date (market data date) to mark the swap to market. One important difference, then, is that the NPV will include all cashflows that fall after the Value Date, rather than those that fall after the Spot Date. (Cashflows that occur on the Value Date will be included if the NPV_INCLUDE_CASH parameter is set to true.)

Formulas

The formula for the Upfront Fee to be paid to the buyer of protection is:

$$UpfrontFee = \left|SpreadDV01(t_{spot})\right| * (Q - c_{Inc}) - \left|A_0(t_{spot})\right|$$

If this quantity is negative, then the cash will be paid to the protection seller instead of the protection buyer.

The NPV on the Value Date is

$$NPV^{Morgan}(t_{ValDate}) = A_0(t_{ValDate}) + SpreadDV01(t_{ValDate}) * (c_{Inc} - Q)$$

In these formulas,

$$SpreadDV01(t_{settle}) = \frac{D(t_1^P)}{c_{Inc}}(C(t_1) - A_0)[p(t_{settle}) - \frac{1}{2}(p(t_{settle}) - p(t_1))]$$
$$+ \frac{1}{c_{Inc}} \sum_{i=1}^{n} D(t_{i+1}^P)\, C(t_{i+1})[p(t_i) - \frac{1}{2}(p(t_i) - p(t_{i+1}))].$$

t_i: Beginning of *ith* accrual period

t_{i+1}: End of *ith* accrual period

t_{settle}: Settlement date, between t_0 and t_1

A_0: Accrual as of settlement date

c_{Inc}: The inception coupon rate of the index

$D(t_i^P)$: discount factor on payment date of *ith* coupon

$p(t_i)$: flat-spread probability of survival from the Value Date to beginning of day on t_i; so $p(ValDate) = 1$.

The coupons are calculated as

$$C(t_{i+1}) = N_{ValDate} c_{Inc} (t_{i+1} - t_i)$$

$N_{ValDate}$: Signed notional of the undefaulted assets as of the Value Date.

The survival probability p(t) is calculated from the quoted market spread s_{Mkt}, which on the Bloomberg CDSW screen is the "Repl Spread." A constant hazard rate (or "clean spread") is defined By

$$h = \frac{s_{Mkt}}{1 - R}$$

and the survival probability at time t is

$$p(t) = \exp(-ht)$$

with *t* computed from the value date in the day count of the market spread s_{Mkt}. Note this is a continuously-compounded curve regardless of the payment frequency of the CDS. (See the "Analytical Benchmarks" section under "Pricing Credit Default Swaps" in this document.)

For a Buy Protection trade, A_0 and the SpreadDV01 will be negative, because premium will be paid out by the buyer; for a Sell Protection trade, they are positive. (The Upfront Fee formula uses their absolute values for clarity.) For a derivation, see the Appendix for this section.

CDSW Comparison

This formulas given above approximate the Bloomberg CDSW screen calculations. A comparison is given in the following table for the ITRAXX 5Y Crossover Index Series 10 Version 1 maturing Dec 2013 and priced on November 26, 2008. The accrued interest is 90,222.22.

Repl Spread	Calypso	Bloomberg
896.81	1,028,231.85	1,028,237.00
749.849	575,082.37	575,047.00
699.861	409,107.82	409,104.00
649.874	236,718.08	236,767.00
599.885	57,638.38	57,774.00
579.89	(15,923.87)	(15,748.00)

Example of Upfront Fee from Quote

The following screenshot shows the CDSW screen for pricing an index from a quote.

Value Date (Curve Date): 12/14/05

Settlement Date: 12/15/05

"Cash Paid On" Date: 12/19/05

The trade is at a quoted spread of 46.249 basis points, versus an inception spread of 45 basis points. The index is quarterly, Act/360, and the trade pays premium, buys protection.

The next table details the calculation. The discount factors are taken from the Bloomberg cashflow page, which can be a source of discrepancy: *the Bloomberg page only displays four decimal places, even though the calculation underlying the screen uses higher precision*. The cashflow amounts and probabilities are calculated in Calypso. The last cashflow differs from Bloomberg, due to the fact that Calypso employs the ISDA convention in which an additional day is added for premium accrual at the end of the trade, while Bloomberg does not.

To check the probability for the cashflow ending on 3/20/08, for example, one has:

Q = 0.004625, R = 40%,

days = 3/20/08 – 12/14/05 = 827

$$p(t) = \exp(\frac{0.004625}{1-.4} * \frac{827}{360}) = 0.982448106$$

The PV column in the table is the sum of the two columns to its left, times the discount factor. Note the special handling for the first cashflow, where only the remaining flow is used for the risky portion. This value will depend on the accrual date, and in the table the values are shown for both the Value Date and the Spot Date (the latter in parenthesis). Summing the PV column gives the Spread DV01. In the table the discount factors are as of the Value Date, this Spread DV01 is also for the Value Date and should be revised by the discount factor to Settle Date, which is estimated at 0.9999.

Then the Upfront Fee calculation is:

Accrual(Spot Date) = -10,625

Unaccrued first flow (Spot Date) =- 625

Sum of PV of flows (unaccrued from spot date, discounted to Val Date) = -198,489.00

Sum of PV of unaccrued flows(Spot date) = -198,489.00/0.9999

=198,508.85

SpreadDV01(Spot Date) = 198,508.85/ 45bp =-4,411.31

*SpreadDV01(Spot Date) * (c_{Inc} – Q) = -4,411.31 * (45 – 46.25) = 5514.14*

*NPV(Spot Date) = SpreadDV01(Spot Date) * (c_{Inc} – Q) + Accrual (Spot Date)*

= 5514.14 - 10,625 = -5,110.86

UpfrontFee = - NPV(Spot Date) = 5,110.86

The calculation of the NPV on the Value Date, which is the NPV Pricer Measure, is (note the adjustment for the greater unaccrued first flow):

Accrual(Value Date) = -10,500

Unaccrued first flow (Val Date) = -750

Sum of PV of flows (Val Date) = 198,489.00

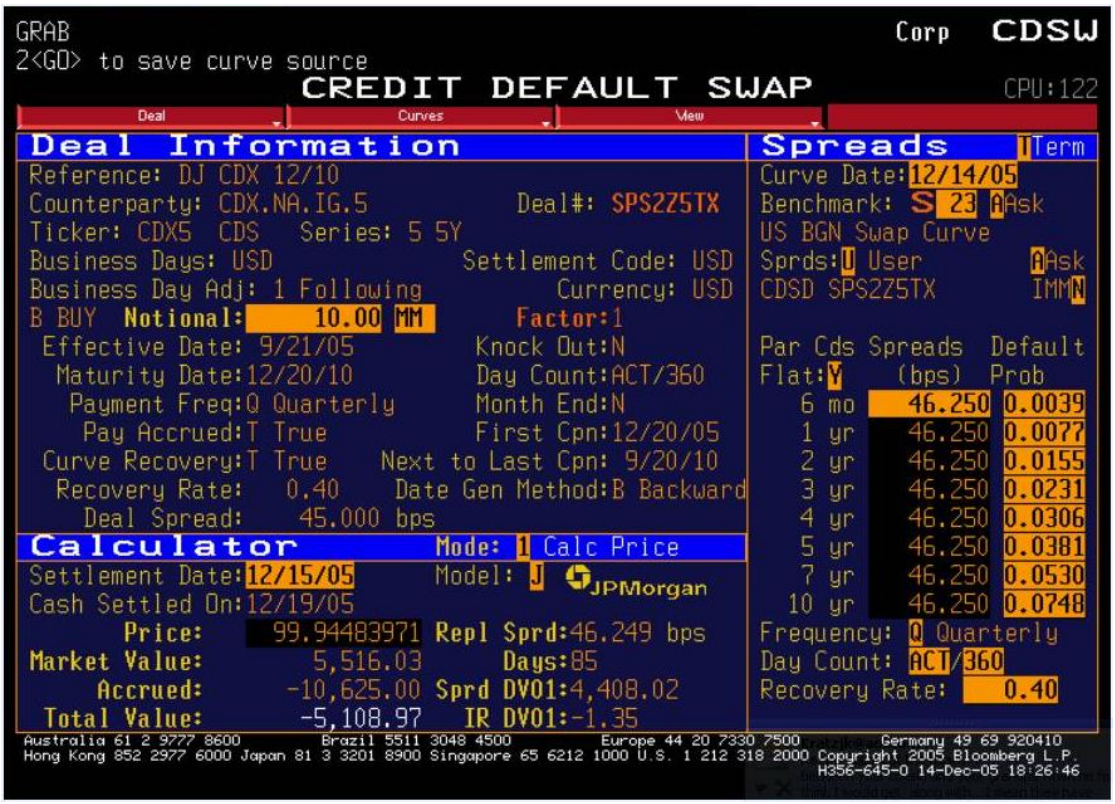
SpreadDV01(Value Date)= 198,489.00/45bp = 4,410.87

*SpreadDV01(Value Date) * (c_{Inc} – Q)= -4,410.87* (45 – 46.25) = 5,513.58*

NPV(Value Date) = 5,513.58– 10,500 = -4986.42

Note: Although the quote is associated with the Spot Date, because it defines a probability curve for the Value Date it is consistent to use that curve when calculating the NPV as of the Value Date.

The Spot Date valuation is very close to the Bloomberg result; the difference is due to the uncertainty in the displayed discount factors and potential differences in the method used to generate the flat-spread probabilities.



Pmt Begin	Pmt End (= Pmt Dt)	Period	Proj Amt (Unacc)	Start Survival Prob.	End Survival Prob.	df	pEnd*C	(pStrt - pEnd) *C/2	PV Prem
9/21/2005	12/20/2005	0.25	625	1	0.9998715	0.9994	624.92	0.04	624.58
12/20/2005	3/20/2006	0.25	11250	0.9998715	0.9979466	0.9883	11,226.90	10.83	11,106.25

Pmt Begin	Pmt End (= Pmt Dt)	Period	Proj Amt (Unacc)	Start Survival Prob.	End Survival Prob.	df	pEnd*C	(pStrt - pEnd) *C/2	PV Prem
3/20/2006	6/20/2006	0.25556	11500	0.9979466	0.9959826	0.9763	11,453.80	11.29	11,193.37
6/20/2006	9/20/2006	0.25556	11500	0.9959826	0.9940226	0.9644	11,431.26	11.27	11,035.18
9/20/2006	12/20/2006	0.25278	11375	0.9940226	0.9920876	0.9527	11,285.00	11.01	10,761.70
12/20/2006	3/20/2007	0.25	11250	0.9920876	0.9901776	0.9415	11,139.50	10.74	10,497.95
3/20/2007	6/20/2007	0.25556	11500	0.9901776	0.988229	0.9302	11,364.63	11.20	10,581.80
6/20/2007	9/20/2007	0.25556	11500	0.988229	0.9862842	0.9191	11,342.27	11.18	10,434.96
9/20/2007	12/20/2007	0.25278	11375	0.9862842	0.9843643	0.9082	11,197.14	10.92	10,179.16
12/20/2007	3/20/2008	0.25278	11375	0.9843643	0.9824481	0.8973	11,175.35	10.90	10,037.42
3/20/2008	6/20/2008	0.25556	11500	0.9824481	0.9805147	0.8865	11,275.92	11.12	10,005.96
6/20/2008	9/22/2008	0.26111	11750	0.9805147	0.9785432	0.8756	11,497.88	11.58	10,077.69
9/22/2008	12/22/2008	0.25278	11375	0.9785432	0.9766383	0.8651	11,109.26	10.83	9,619.99
12/22/2008	3/20/2009	0.24444	11000	0.9766383	0.9747998	0.855	10,722.80	10.11	9,176.64
3/20/2009	6/22/2009	0.26111	11750	0.9747998	0.9728398	0.8443	11,430.87	11.52	9,660.80
6/22/2009	9/21/2009	0.25278	11375	0.9728398	0.970946	0.8341	11,044.51	10.77	9,221.21
9/21/2009	12/21/2009	0.25278	11375	0.970946	0.969056	0.8239	11,023.01	10.75	9,090.72

Pmt Begin	Pmt End (= Pmt Dt)	Period	Proj Amt (Unacc)	Start Survival Prob.	End Survival Prob.	df	pEnd*C	(pStrt - pEnd) *C/2	PV Prem
12/21/2009	3/22/2010	0.25278	11375	0.969056	0.9671696	0.8138	11,001.55	10.73	8,961.80
3/22/2010	6/21/2010	0.25278	11375	0.9671696	0.965287	0.8038	10,980.14	10.71	8,834.44
6/21/2010	9/20/2010	0.25278	11375	0.965287	0.9634079	0.7939	10,958.77	10.69	8,708.65
9/20/2010	12/20/2010	0.25556	11500	0.9634079	0.9615326	0.7841	11,057.62	10.78	8,678.74

Appendix - Derivation of Price from Quote

The price from quote formula is derived using the assumptions of flat spread and 40% recovery as described previously.

The premium payments are paid at the inception premium rate of the CDS index. Usually a CDS index will be of the "Pay Accrual" type, that is, premium will accrue on existing notional up to each credit event, when the notional will decrease. At the beginning of each day, it will be assumed that premium will accrue on the remaining (undefaulted) notional as of the start of the day; if there are credit events during the day, these will be assumed to affect the next (business) day's notional.

In pricing, one can project an expected remaining notional for any future date. Then one can define an effective survival probability with respect to the value date as follows.

$$p(t) = \frac{\text{Expected remaining Notional at t}}{\text{Current Notional as of Value Date}}$$

Given this effective probability, the NPV of the index swap can be computed in the same manner as a single-name CDS. The one difference is the interpretation of the probability as remaining notional fraction, which means the accrued premium could have one or more changes of notional in each coupon period. The protection seller receives the coupon for the entire period, even if the trade was settled after the period start. For a CDS paying inception coupon c_{Inc} with time-dependent expected notional N(t), the total expected amount of the received premium coupons to maturity T is therefore:

$$E[A] = \int_{t_0}^{T} N(t)\, c_{Inc}\, dt$$

Here, the lower limit t_0 is the start of the accrual period for the next cashflow to be received.

This expression is applicable for aged trades, in which t_0 is less than or equal to the Settlement Date (up to amortizing for past credit defaults), as well as forward-starting trades in which t_0 is after the Settlement Date. In the following, the case of the aged trade will be considered. (The forward-starting case is readily obtained by setting the accrual to zero.)

Assume the Settlement Date, t_{settle}, is either the Value Date or the next business day (Spot Date). In this case, the accrued portion of the current coupon is known and riskless (the accrual to the next business day is known from the notional as of the beginning of the Value Date). If the swap has notional $N_{ValDate}$ on the Value Date and otherwise is not amortizing, one has:

$$E[A] = A_0 + \int_{t_{settle}}^{T} p(t) N_{ValDate}\, c_{Inc}\, dt$$

and the discounted expected value of the premium payments as of the Settlement Date is

$$NPV_{premium}(t_{settle}) = D(t_1^P)[A_0 + \int_{t_{settle}}^{t_1} p(t) N_{ValDate}\, c_{Inc}\, dt)] + \sum_{i=1}^{n} D(t_{i+1}^P) \int_{t_i}^{t_{i+1}} p(t) N_{ValDate}\, c_{Inc}\, dt$$

> t_i: Beginning of *ith* accrual period
>
> t_{i+1}: End of *ith* accrual period
>
> t_{settle}: Settlement date, between t_0 and t_1
>
> A_0: Accrual as of settlement date
>
> c_{Inc}: The inception coupon rate of the index
>
> $N_{ValDate}$: Notional of the undefaulted assets as of the Value Date
>
> $D(t_i^P)$: discount factor on payment date of *ith* coupon
>
> $p(t_i)$: probability of survival from the Value Date to beginning of day on t_i;
>
> so *p(ValDate) = 1*.

One fine point is the separation between the known, risk-free accrual and the remaining risky part. The accrued amount A_0 accrues on the notional as of the start of day on the Value Date, and so is known if Settlement Date is just one business day later. Thus it is taken as a guaranteed (risk-free) amount of the coupon. Then the swap begins accruing there may already be a reduced notional on the first day, due to credit events that might have occurred between the Value Date and the Settlement Date. The probability curve is created as of the Value Date so that this effect is captured.

The evaluation of this expression can be performed as described in the section on single-name CDS. Using the Morgan approximation, the survival probability curve is approximated as linear over each coupon period:

$$p(t) = p(t_i) - h_i(t - t_i), \quad t_i <= t <= t_{i+1},$$

The slope is determined by the endpoints of the period:

$$h_i = \frac{p(t_i) - p(t_{i+1})}{(t_{i+1} - t_i)}$$

Then one finds

$$\int_{t_i}^{t_{i+1}} p(t)dt = (t_{i+1} - t_i)[p(t_i) - \frac{1}{2}(p(t_i) - p(t_{i+1}))]$$

Written in this form, the two contributions can be interpreted as the notional of the coupon if there are no further defaults, minus the average expected decrease in notional from defaults during the coupon period.

Define the amounts

$$C(t_{i+1}) = N_{ValDate} c_{Inc} (t_{i+1} - t_i)$$

which are the coupon payments that would be accrue as of t_{i+1} if there were no defaults after the value date. Then the present expected value of the premium payments as of the Settlement Date becomes:

$$NPV_{premium}(t_{settle}) = D(t_1^P)[A_0 + (C(t_1) - A_0)(p(t_{settle}) - \frac{1}{2}(p(t_{settle}) - p(t_1)))]$$
$$+ \sum_{i=1}^{n} D(t_{i+1}^P) C(t_{i+1})[p(t_i) - \frac{1}{2}(p(t_i) - p(t_{i+1}))]$$

Since A_0 is certain to be paid, it is conventionally separated from the remainder of the NPV, which is regarded as "risky." One writes:

$$NPV_{premium}(c_{Inc};\ t_{settle}) = D(t_1^P)A_0 + SpreadDV01(t_{settle}) * c_{Inc},$$
$$SpreadDV01(t_{settle}) = \frac{D(t_1^P)}{c_{Inc}}[(C(t_1) - A_0)(p(t_{settle}) - \frac{1}{2}(p(t_{settle}) - p(t_1)))]$$
$$+ \frac{D(t_1^P)}{c_{Inc}} \sum_{i=1}^{n} D(t_{i+1}^P)\ C(t_{i+1})[p(t_i) - \frac{1}{2}(p(t_i) - p(t_{i+1}))].$$

The *Spread DV01* is also known as the *Risky PV01* or the *Risky Duration*. It is the present expected value of one basis point of premium, *for fixed default probabilities p(t)*, of a CDS that begins on the Settlement Date. As it only includes unaccrued premium, the first cashflow in this expression is only the unaccrued portion, $C(t_1 - A_0)$, of the actual CDS index cashflow.

To find the Upfront Fee, suppose the quoted spread is Q. The NPV of the premium leg will depend on Q through the flat-spread probability curve, *p(t),* constructed from Q (see below).

The NPV of the protection leg is found from the specification that the quoted spread Q is the breakeven rate for the hypothetical CDS starting on the Spot Date. By definition, the breakeven rate is the premium rate that gives zero NPV for a Settlement Date equal to the Spot Date. Write this as:

$$NPV'_{protection}(t_{spot}) + NPV'_{premium}(Q; t_{spot}) = 0$$

The primes on the NPV indicate this is that hypothetical swap that starts accruing on the Spot Date. The NPV of a premium leg starting on the Spot Date with spread Q is just the Spread DV01 found above (present value of 1 bp) times Q. Thus:

$$NPV_{protection}(t_{spot}) = -SpreadDV01(t_{spot}) * Q$$

Summing the protection and premium NPVs gives:

$$NPV(t_{spot}) = D(t_1^P)A_0(t_{spot}) + SpreadDV01(t_{spot}) * (c_{Inc} - Q).$$

The date for the accrual, t_{spot}, is here made explicit.

There is one more assumption in the Morgan approximation, which is that the accrued premium A_0 does not need to be discounted. This is equivalent to assuming that if a default event occurred today, the premium already accrued would be paid immediately, rather than delayed to the usual premium payment date. (Note this differs from the usual assumption of delayed payment that Calypso makes for pricing CDS from probability curves with NPV_FROM_QUOTE equal to False.) Thus:

$$NPV^{Morgan}(t_{spot}) = A_0(t_{spot}) + SpreadDV01(t_{spot}) * (c_{Inc} - Q).$$

The Upfront Fee is the payment that will cause the NPV with fee included to be zero, so:

$$UpfrontFee = -[A_0(t_{spot}) + SpreadDV01(t_{spot}) * (c_{Inc} - Q)].$$

For a Buy Protection trade, A_0 and the Spread DV01 will be negative, because premium will be paid out by the buyer; for a Sell Protection trade, they are positive.

As noted previously, the NPV on the Value Date is computed by setting the Settlement Date in the above formulas to the Value Date.

$$NPV^{Morgan}(t_{ValDate}) = A_0(t_{ValDate}) + SpreadDV01(t_{ValDate}) * (c_{Inc} - Q)$$

It should be noted that in the Bloomberg screen the discount curve is the Euro Swap Curve (EU BGN Swap Curve), but Calypso will use whatever discount curve the user has configured for the index product.

9.4 CDS Index Basis Adjustment

Pricing a CDS Index using probability curves does not generally give the same result as pricing from a market quote. There are two reasons for this. One is the difference in liquidity between trading an index and trading the credit default swaps on the individual names that equate to the index. The other is the market convention used to quote a CDS index spread and calculate an upfront fee, which assumes a flat spread curve rather than real market survival curves.

The difference between the breakeven spread calculated from probability curves and that quoted in the marketplace is called the *CDS index basis*.

It is possible to alter the probability curves so that the breakeven spread they produce is equal to the market quoted spread. This functionality is offered in Calypso in the following form. Define a *CDS Basis Adjustment* by a single number, *h*, which is regarded as a constant hazard rate. For every survival probability curve P_k used in index pricing, multiply each stored survival probability $P_k(t_i)$ by a factor:

$$P_k(t_i)^{adj} = \exp(-ht_i)P_k(t_i)$$

The t_i are the act/365 times to each stored point of the curve. To find the survival probability at any other times, interpolation is done on the adjusted probabilities. That is, after the above adjustment, the resulting curve is used by the pricer as though it were the originally input probability curve. Note that *h* depends only on the index, and not on each reference asset.

The Pricing Parameter **USE_CDS_BASIS_ADJ** determines whether or not this adjustment will be applied to the curves before pricing. This parameter is available only in index pricers: `PricerCDSIndex` and `PricerCDSIndexTranche`. If the parameter is set to False, the unadjusted probability curves for each reference asset will be used. If set to True, the pricer will look up the stored CDS Basis Adjustment for the index of the trade and create adjusted curves as described. These are used to calculate pricer measures as usual. The adjusted curves are not stored, but exist only for the duration of their use by the pricer. When all has been calculated, they are discarded.

The CDS Basis Adjustment for an index -- the value of *h* -- is stored in the CDS Basis Adjustment Curve associated with that index.

9.5 CDS Index Tranche Valuation

CDS Index Tranche trades are valued as though they were straightforward CDSNthLoss (CDO) trades. There are a few differences. One is that the next premium coupon of an index tranche trade is paid in full, even if the tranche is bought or sold in the middle of a coupon accrual period. Another is that the CDS Basis Adjustment can be applied to the probability curves if the user chooses, in order to correct for the difference between trading an index and trading the underlying assets.

To price the trade on a given value date, the `PricerCDSIndexTranche` performs the following steps:

1. Converts the CDS Index Tranche trade to a CDSNthLoss trade whose start date is equal to the start date of the premium coupon period in which the value date falls. The reference basket of the trade is the index basket.
2. Obtains an associated CDSNthLoss pricer.
3. If USE_CDS_BASIS_ADJ is True, the pricer obtains the CDS Basis Adjustment from the CDS Basis Adjustment Curve of the CDS Index, and adjusts the probability curves accordingly.
4. The CDSNthLoss trade is priced as usual.

The pricer that is used to evaluate the CDSNthLoss trade is one of the following:

- If using `PricerCDSIndexTranche`, the pricer that is used for the CDSNthLoss trade is that which is associated in the PricerConfig with the CDSNthLoss product. This is chosen by the user.
- If using `PricerCDSIndexTrancheOFM`, the one-factor model pricer will be used, that is, PricerCDSNthLossOFM.
- The user can specify (through code) any other CDSNthLoss pricer to use for an index trade. The user needs to subclass `PricerCDSIndexTranche` and override the method getNthLossPricer so that it returns the desired pricer.

Usually the one-factor model is used to price CDS Index Tranche trades, because the implied correlations are readily available from the marketplace for new trades. From the market quotes, a Base Correlation Surface can be created from which aged CDS Index trades can be valued. The following section describes the construction and use of the Base Correlation Surface.

Section 10. Base Correlation Surfaces

10.1 Base Correlations and Correlation Skew

From market quoted spreads for index tranches one can solve for the implied correlation that produces that spread. That correlation can then be used to value other trades with similar tranches and baskets. For the Gaussian copula one-factor model, or its Monte Carlo equivalent, there should theoretically be only one correlation that prices every tranche. However, in reality the market employs a different correlation for each tranche level. This phenomenon is the *correlation skew*.

Because of the skew, for trades on similar baskets but with very different tranches, finding the correlation to use as input presents a problem. For example, an index tranche of 7-10% can, over time, pay out losses so that it is reduced to, for example, an effective 5-8% tranche. When the market quotes 0-7% and 7-10%, it is not clear how to derive the off-the-run tranche correlation.

The *base correlation* concept provides a means to evaluate any tranche on a given basket. A base correlation is the implied correlation for a tranche that begins at 0% and has detachment point of X%, where X is any convenient value. A *base correlation surface* is a set of base correlations at different detachment points for a given index tenor; there can be more than one index tenor in a surface. From this information, one can value any tranche on that index basket.

Let NPV(A,B) be the NPV of a tranche with attachment A and detachment B. In terms of base tranches,

$$NPV(A,B) = NPV(0,B) - NPV(0,A)$$

which follows from the additive nature of expected losses. Thus one needs the implied correlations for the 0 to A and the 0 to B tranches, which can be obtained from a base correlation surface (supposing one has been created). If the A and B detachment points are not explicitly present in the base correlation surface, they can be interpolated. Performing two NPV calculations, one with each correlation and tranche, and taking the difference provides the desired result.

Note that calculating NPV using base correlations takes twice as long as using a single tranche correlation. For this reason it is usually preferable to employ the faster one-factor model when using base correlations rather than Monte Carlo. The standardized nature of CDS index trades also makes them amenable to the one-factor model.

Using the same NPV technique and given a set of index tranche market spreads, one can iteratively solve for the base correlations that reproduce those spreads.

Reference:

"Credit Correlation: A Guide" JP Morgan Credit Derivatives Strategy (March 2004)

10.2 Generation of Base Correlations

10.2.1 Market Data Inputs

Given the market break-even rates for CDOs on an index, the goal is to find the base correlations. An example of market data is:

17 May 2005	level (bp*)
DJ CDX NA IG 4	

Series	
Full index	78.0
0-3% *	61.88%
3-7%	281.5
7-10%	67.0
10-15%	35.5
15-30%	20.5

*All 0-3% tranches are quoted as percentage upfront plus 500bp running premium.

In this case, the base correlations are sought for the following tranches:

0-3%

0-7%

0-10%

0-15%

0-30%

The procedure for generating the base correlations is defined by the manner in which they will be subsequently used by the pricers. Let there be a CDO with attachment point A (lower tranche level) and detachment point B (upper tranche level). It will be supposed that the pricer will compute the NPV of this CDO as follows.

10.2.2Pricing a Tranche from Base Correlations

1. Look up the base correlation for 0-A. (This may require interpolation.)
2. Use this correlation to find the NPV the CDO as if it had a 0-A tranche. Call this NPV(0,A).
3. Look up the base correlation for 0-B.
4. Find the NPV of the 0-B tranche. Call this NPV(0, B).
5. Return the difference: NPV(0,B) – NPV(0,A).

It is only necessary to reverse this procedure to generate base correlations when given breakeven rates. The NPV of each market CDO is already known that, as a trade priced at its breakeven rate has, by definition, NPV equal to zero. Define

NPV(A, E; r, c): Net present value of A-E tranche with premium rate *r* and correlation *c*, not including any upfront fee.

r_i: Market-quoted breakeven rate for the *ith* tranche

c_i: Base correlation for *ith* base tranche

E_i: detachment point of *ith* tranche of the market data

In the above example, E_1 = 3%, E_2 = 7%, E_3 = 10%, an so on.

For each CDO in the market data, the implied compound correlation c_I is the correlation that gives 0 NPV, including fees, for the given breakeven rate r_{BE}:

$$NPV(0, E_1; r_{BE}, c_I) + UpfrontFee = 0$$

To find base correlations, the implied correlations are not needed (other than for the first tranche, where it is the same as the base correlation). The procedure for finding base correlations from quotes is described below.

10.2.3 Generating Base Correlations from Breakeven Prices

1. Start with the equity tranche, which has attachment point 0%. These tranches are priced with an upfront fee as well as a premium rate. Solve for the correlation c_1 for this tranche

$$NPV(0, E_1; r_1, c_1) = -(E_1 Upfront\ Fee)$$

2. The NPV of the next tranche, (E_1, E_2), is desired to be the difference between the NPVs found for the base tranches, when the quoted breakeven rate r_2 is used as the premium rate. For some implied correlation c_{I2}, one has:

$$NPV(E_1, E_2; r_2, c_{I2}) = NPV(0, E_2; r_2, c_2) - NPV(0, E_1; r_2, c_1)$$

But the implied correlation is not needed, because by definition the NPV at the breakeven rate is 0, as there is no upfront fee for this tranche. So the equation reduces to:

$$NPV(0, E_2; r_2, c_2) = NPV(0, E_1; r_2, c_1)$$

This equation is solved for c_2. To do so, first the NPV on the right hand side need to be found: note this uses the first tranche, (0, E_1), and the first base correlation c_1, but the breakeven rate of the second tranche, r_2.

3. Repeat for each additional tranche. To find the *nth* base correlation using the nth breakeven rate, first find the NPV of the previous base tranche using the previous correlation and current breakeven rate: calculate

$$NPV(0, E_{n-1}; r_n, c_{n-1})$$

Then solve for c_n so that:

$$NPV(0, E_n; r_n, c_n) = NPV(0, E_{n-1}; r_n, c_{n-1})$$

Using the pricers, this can be done by creating a Fee equal to the right hand side of this equation, and then solving for the breakeven correlation for the (0, E_n) tranche with premium rate r_n.

10.2.4 Super-Senior Tranches

A *super-senior tranche* is a tranche that has a detachment of 100%: for example, the 30-100% tranche. Theoretically, the breakeven premium rate for the super-senior tranche should be derived from the other tranches up to 30% and the full index (0 to 100% protection):

$$\text{ExpLoss}(30\%, 100\%) = \text{ExpLoss}(0\%, 100\%) - \text{ExpLoss}(0\%, 30\%)$$

In the absence of arbitrage, this must hold. However, in reality arbitrage opportunities do occur in the marketplace, and the super-senior tranche can have a market quote that differs from its theoretical rate.

Calypso will generate a base correlation for the super-senior tranche if a quote is given for this tranche. This correlation is just the flat (implied) correlation that produces the quoted rate as the breakeven rate.

When pricing a super-senior tranche trade, Calypso will use the flat correlation for that tranche if it exists on the correlation surface. If no such correlation exists, as is the case when no quote was entered for the super-senior tranche, then Calypso will price the super-senior tranche using the base correlation at the detachment point and the no-arbitrage assumption given above.

10.3 Extrapolation in Time

Suppose the earliest market quote used in generating an index correlation surface is the 5 year maturity. To obtain a correlation for a trade with a 3 year maturity, an extrapolation method must be used. This is done with the "Moneyness Mapping" method (described in detail in the section on Bespoke Basket Correlation). The following describes the steps.

Calculate:

EL(5Y) = Expected Loss of a 5Y CDS trade on the underlying whole index.

EL(3Y) = Expected Loss of a 3Y CDS trade on underlying whole index.

Define:

$$W = \frac{EL(5Y)}{EL(3Y)}$$

From the market quotes, the correlation of each detachment point at the 5Y tenor have been generated. Denote

Corr(D, 5Y) = Correlation of base detachment D for 5Y tenor

The correlations at the 3Y for the same detachment is defined as:

$$Corr(D, 3Y) = Corr(D * W, 5Y)$$

That is, the factor W is used to shift the detachment point, the correlation at that detachment is found at the 5Y point (using the detachment interpolator on the surface).

For example: To find the correlation for the 7% detachment point, one computes 7% * W to get a 5Y detachment point. Suppose it is 8.1%. Then interpolate at the 5Y tenor to find the correlation at 8.1%. Suppose one finds 0.67. Then one sets the correlation at the 7% detachment point for 3Y tenor to be 0.67.

In generating a surface, a user can specify a set of date in the SPECIFIC_DATES parameters (separated by commas) and the extrapolation will be done on these dates.

Section 11. Bespoke Basket Correlations

11.1 Introduction to Bespoke Mapping

Bespoke CDO tranche trades are over-the-counter trades that depend upon an agreed-upon baskets of reference entities, rather than a published index basket. ("Bespoke" is used in the sense of "custom-made" or "personalized.")

Base correlation surfaces are a solution to the problem of finding the market correlation to use in pricing an off-market index tranche. But how is an index correlation surface to be used for a non-index basket?

A mapping is needed to associate an index correlation surface (or set of surfaces) with a bespoke basket. There are clearly a number of ways of doing this; for example, one might use historical correlations between the entities in the bespoke basket and the entities in the index basket. However, this begs the question of the source of those correlations. Part of the value of index tranche trading is the ability it provides to read correlations from the marketplace.

A procedure growing is popularity is the use of a *mapping measure* (or *risk measure*) to provide a means of comparing index and bespoke baskets. This measure can be defined in various ways, but usually it would involve valuing the losses that the portfolio is expected to produce over the trade lifetime. The idea is that correlations can be associated directly with certain risk values, independent of the reference basket composition. This measure then is an alternative way of expressing the correlation surface.

Calypso offers two methods for performing this mapping.

The first is the *Expected Loss Method* that calculates the expected losses of the index tranches on the one hand, and the bespoke tranche on the other using an estimate for the bespoke correlation. By comparing these losses the bespoke correlation estimate is improved.

The second is the *Moneyness Method*, which is faster and simpler. The expected losses of the whole baskets for the index and bespoke trades are compared; since these are independent of correlation, there is no need to make an initial correlation estimate. The ratio of these basket losses is used to modify the detachment point at which the correlation is looked up using the index surface.

Whichever method is used, the mapping can be done either by generating a separate Bespoke Correlation Surface for each bespoke basket, or more simply by performing the mapping from an Index Correlation Surface on demand during trade pricing.

References:

A. Reyfman, K. Ushakova, W. Kong, "How to Value Bespoke Tranches Consistently with Standard Ones", Bear Stearns Credit Derivatives Research (Sept 23 2004)

J. Turc, D. Benhamou, B. Herzog, M. Teyssier, “Pricing Bespoke CDOs: Latest Developments”, Societe Generale Credit Research (July 28 2006)

11.2 Expected Loss Method

An index correlation surface is chosen as the starting point. The correlations and the mapping measure for each base tranche should already be computed and stored in the surface. In order to map with the Expected Loss measure, for example, the surface needs to have been generated with the parameter CALC_EXPECTED_LOSS set to True.

As each detachment point on the surface corresponds to a correlation and a mapping measure value, there is a function $S_I(M)$ that associates a mapping measure value M to a correlation:

$$\text{Index Surface Corrrelation at M} = S_I(M)$$

Interpolation or extrapolation defines a correlation for any M, provided that M is monotonic, i.e., it must only trend in one direction, so that there is a unique correlation for each M value.

Given a trade on a bespoke basket tranche with attachment 0 and detachment D_B, the mapping measure for this trade can be computed for any correlation c:

$$M_B(D_B, c) = \text{Mapping measure value for Bespoke Tranche 0 - } D_B \text{ at correlation c}$$

There will be some correlation that produces the same mapping measure value for the bespoke tranche trade and some base tranche on the index surface. That correlation satisfies:

$$S_I(M_B(D_B, c')) = c'$$

This correlation *c'* is then assigned to the bespoke surface at detachment point D_B. Repeating this procedure for all desired detachments of the bespoke basket results in the *bespoke base correlation surface*. These bespoke correlations are applied to the pricing of trades on the bespoke basket, including tranches that are not base tranches.

A numerical solver (based on Newton-Raphson) is employed to find the *c'* such that this equation is satisfied. This value can be corroborated by the user by means of a trade on the bespoke basket tranche. Manually enter correlations into the trade pricing parameters tab and evaluate the associated mapping measures (e.g., the EXP_LOSS pricing measure). Compare these measures against the index surface until one finds a correlation that produces the same measure value in both the bespoke trade and the index surface. The resulting value should be the same (within some tolerance) as the *c'* found by the Calypso bespoke correlation generator.

Details of Expected Loss Interpolation

A correlation surface usually has multiple tenors. Which one is used for mapping a bespoke trade of a particular maturity? The answer is that Calypso interpolates between tenors. The index surface has expected losses stored for each tenor and each detachment point. When looking up a given value of expected loss E, Calypso takes the maturity date T_M of the bespoke trade and interpolates the expected losses at T_M, for each detachment point. Then it searches across these T_M-tenored expected losses to find the detachment point that corresponds to E, using interpolation across detachment points as needed. If there is more than one detachment point with the same E, the point closest to the bespoke tranche is taken.

Expected Loss Mapping Measures

The measure used for mapping index baskets to bespoke baskets can be defined in several ways. The following methods make use of the present value of the probability-weighted loss amounts of a tranche or basket, which is just the NPV of the protection leg.

The simplest measure is the expected loss of the tranche expressed as a fraction of the notional of the basket:

$$ExpectedLoss = \frac{PV\ Expected\ Tranche\ Loss}{CurrentBasketNotional}$$

Another possibility is to attempt to to reduce the dependence of the measure on the probability curves and the maturity of the trade. This measure divides the expected tranche loss by the expected loss of the entire basket:

$$ExpectedLossRatio = \frac{PV\ Expected\ TrancheLoss}{PV\ Expected\ Basket\ Loss}$$

The default mapping measure in Calypso is ExpectedLoss. Software developers can introduce new measures by overriding the computeMappingMeasure method in PricerCDSNthLoss.

11.3 Moneyness Method

The Moneyness Method is based on the simple ratio of expected loss on the bespoke versus the index basket. It is a fast method as it does not depend on making an initial correlation estimate and iterating for a solution. The method finds the index base tranche that has the same *moneyness* as a given bespoke base tranche, where moneyness is defined as the ratio of the detachment point to the expected loss of the whole basket.

Define:

EL(Basket): present value of the expected loss of the whole basket. Calculated as the NPV of the credit leg of a credit default swap on the basket, covering 0 to 100% loss.

Then the Moneyness Method performs the following steps:

1. Computes the EL value for both the index basket and the bespoke basket, using the same notional for each basket, and then calculate the ratio:

$$W = \frac{EL(IndexBasket)}{EL(BespokeBasket)}$$

2. For a trade on the bespoke basket for a base tranche with detachment B (attachment 0), one then finds a corresponding index detachment point:

$$B_{Index} = B * W$$

3. Finally use the Index Correlation Surface to look up the correlation at the index detachment point B_{Index}. This gives the base correlation to use for the bespoke tranche:

$$BaseCorrelationBespoke(B) = BaseCorrelationIndex(B_{Index})$$

11.4 Mapping from Multiple Surfaces: Correlation Formula

Mapping can be done from more than one index surface. Each surface is mapped separately, then combined using weights that sum up to one. The weights are defined by the user in a Correlation Formula.

Suppose the index surfaces are labeled i = 1, 2,..N and a Correlation Formula has been defined with weights a_i,

$$\sum_{i=1}^{N} a_i = 1, \quad 0 <= a_i <= 1$$

For the *ith* surface, the correlation for a tranche B on a bespoke basket can be found using one of the mapping methods; call this correlation BaseCorrelationBespoke(B, i). Then the end result of the mapping from multiple surfaces is

$$BaseCorrelationBespoke(B) = \sum_{i=1}^{N} a_i * BaseCorrelationIndex(B, i)$$

11.5 Usage: Pre-Generation Versus Mapping on Demand

11.5.1 Single Surface

There are three methods in Calypso to value a bespoke trade. These methods depend on how the user has established the correlation surfaces, and on the setting of the parameter ALLOW_BESPOKE_MAPPING.

First Method: Use any correlation surface without mapping

The simplest technique is to assume that an already-existing correlation surface provides a good enough estimate for a particular bespoke basket. The Pricer Config should be set up so there is an entry associating the bespoke basket with that correlation surface. The parameter ALLOW_BESPOKE_MAPPING must be set to False. Then no mapping is done and correlations are just read directly from the given surface. In summary, the Pricing Environment should be set as follows:

- Correlation Surface associated with bespoke basket: Any
- ALLOW_BESPOKE_MAPPING: False

Second Method: Index surface using on-the-fly mapping during pricing (Allow_Bespoke_Mapping = True)

If one wants to use bespoke-corrected correlations, the method that requires the least preparation on the part of the user is to do the correlation mapping at the time the bespoke pricing is performed. The Pricer Config needs to have an entry associating the bespoke basket with an index correlation surface. At the time of pricing, bespoke base correlations will be drawn from the index surface using the mapping method of the preceding section. For a general tranche, two base correlations will be needed, one each for the attachment and detachment points, and hence two mapping procedures will be performed.

The mapping procedure needs to be activated. To do this, the parameter ALLOW_BESPOKE_MAPPING must be set to True.

In summary, the Pricing Environment should be set as follows:

- Correlation Surface associated with bespoke basket: Any that has expected loss generated on it (CALC_EXPECTED_LOSS set to True when surface was generated).
- ALLOW_BESPOKE_MAPPING: True

Third Method: Two steps --(1) Generate a Bespoke correlation surface (2) Use when pricing

The third method is to generate a correlation surface specifically for the bespoke basket, doing the mapping at the time of generation. Then pricing is done using this surface. This saves time during pricing by doing all the correlation mapping ahead of time. The inconvenient aspect is a proliferation of correlation surfaces in the database.

To generate a bespoke correlation surface, one must select an already-generated index correlation surface to use as the source. If the Iterative Method is desired, the index surface needs to have been generated with CALC_EXPECTED_LOSS set to True, as the expected loss is used in the Iterative mapping.

In the correlation surface window, select the bespoke generator. To use the "Expected Loss Method", select the generator **BespokeCorrelation**; to use the "Moneyness Method", select **BespokeCorrelationMoneyness**.

For the parameter SURFACE_NAME, enter the name of the source index correlation surface. In the Basket field, select the Bespoke basket Generate and save. Then set the Pricer Config so this bespoke surface is associated with the bespoke basket.

Because the pricer now does not need to do correlation mapping, set ALLOW_BESPOKE_MAPPING to False. Otherwise, one would end up doing mapping on already-mapped correlations.

In summary, to use a Bespoke correlation surface the Pricing Environment should be set as follows:

- Correlation Surface associated with bespoke basket: Bespoke correlation surface
- ALLOW_BESPOKE_MAPPING: False

11.5.2Multiple Surfaces

On-the-fly mapping during pricing (PricerCDSNthLossBespoke)

A special OFM pricer, PricerCDSNthLossBespoke, will always perform on-the-fly mapping of multiple index surfaces during pricing. No parameter needs to be set (unlike the single-surface case, which uses ALLOW_BESPOKE_MAPPING rather than a separate pricer). After mapping, the regular OFM pricing is performed.

The user needs to create a Correlation Formula and associate it to the desired basket in the Pricer Config. It is this assigned Correlation Formula that the pricer will use for trades on that basket.

Two steps --(1) Generate a Bespoke correlation surface from a Correlation Formula (2) Use when pricing

In the CorrelationGeneratorBespoke, multiple surfaces will be used in the mapping if a Correlation Formula name is provided. Note there are two parameters that are mutually exclusive:

- either enter an "INDEX SURFACE NAME" – which is the name of an existing correlation surface from which the bespoke surface will be derived;
- or enter a "CORRELATION FORMULA" -- the name of an existing Correlation Formula which in turn points to a set of index surfaces.

Once a surface has been generated with a Correlation Formula, it can be used with any pricer, not just the OFM. One should NOT use PricerCDSNthLossBespoke in this case.

Section 12. Pricing of Funded CDS and CDS Baskets

12.1 Single-Name CDS

A funded CDS removes the risk that the Protection Seller will not be able to make payments in case the reference asset defaults, by requiring the Seller make an initial payment equal to the full notional of the CDS. Thus any possible default payment is made in advance.

If there are is no default during the life of the CDS, the Protection Seller receives the full notional back at maturity.

If there is a default, the Protection Seller receives the defaulted asset, if the CDS has physical settlement, or the market recovery value of the asset if the CDS is cash settled. The CDS is then terminated. The net amount the Protection Seller has paid in this case then is the original notional payment minus the recovery value. This is the same net payment as with an unfunded CDS; the only difference is the timing of the cashflows.

Summary of the Cashflows

Definitions:

N: Notional amount of the CDS

R: Recovery rate

Cashflow sequence:

1. Protection Seller initially pays N.

2. Protection Seller receives premiums on N.

3. a) If no default: Protection Seller receives back N.

b) If default: Protection Seller receives recovery R (or physical equivalent), to make up for overpayment of loss amount.

Note that, for the life of the CDS, if there is no default the net cash outflow from the Protection Seller is 0; if there is default, it is (1-R)N.

12.1.1 NPV of Funded CDS

The calculation of a funded CDS is related to an unfunded CDS as follows:

The NPV of the premium coupons is the same as the unfunded CDS.

The NPV of the credit default payments differs in the amount of each payment upon default: for an unfunded CDS it is 1-Recovery, while for a funded CDS it is –Recovery. Note the sign, indicating that it is the Protection Seller who *receives* a payment if default occurs. This is the opposite direction of default payment as compared with an unfunded CDS, due to the fact that the Protection Seller has already overpaid the default in advance.

There is an additional NPV of the principal flows:

$$NPV_{prin} = \sum_{k=1}^{M} D({t_k}^P) S(t_k) N_k$$

with (signed) principal N_k paid on date t_k, if t_k is in the future and the asset survives with probability $S(t_k)$; $D(t_k)$ is the riskless discount factor. A funded CDS with no amortization has two flows, equal in magnitude and opposite in direction, at the start and end of the CDS contract.

12.1.2Breakeven Rate of a Funded CDS: Price from Curve

For a CDS that is not funded, the breakeven rate is defined as the premium spread that produces a zero NPV. If this spread were used to create an unfunded CDS the resulting NPV would be far from zero and of the same order as the notional amount.

For a funded CDS, then, the definition of the breakeven rate needs to be modified. Calypso uses the following definition:

$$NPV(BreakevenRate) = \sum_{k=1}^{M} N_k$$

That is, for a CDS with spread equal to the "Breakeven Rate", the NPV is the sum of the future principal flows. For a funded CDS before the start date, with both principal flows in the future, the right hand side is zero; for an aged CDS, with only the maturity principal flow in the future, the right hand side is the notional amount of the CDS. This is equivalent to pricing a bond at par.

The breakeven rate can be expressed in a general form using the "Spread DV01" (also known as "RiskyPV01" or "Risky Duration"), which is the change in NPV of the CDS if the premium spread were to be increased by 1 basis point (without a change in probability curve). The Spread DV01 is the same for funded and unfunded CDS, as the default and principal payments do not depend on the premium spread. For a CDS with constant fixed rate *s*,

$$NPV_{credit} + NPV_{prin} + SpreadDV01 * s = NPV$$

By definition of the breakeven rate *b*, require:

$$NPV_{credit} + NPV_{prin} + SpreadDV01 * b = \sum_{k=1}^{M} N_k$$

Subtracting one expression from the other gives the result

$$b = s + \frac{\sum_{k=1}^{M} N_k - NPV}{SpreadDV01}$$

This can be also shown to be the correct breakeven spread for a CDS with a floating rate premium of fixed spread *s* to an index.

12.2 Funded CDS Baskets

The cashflows of a funded CDS basket are as follows:

1. Protection Seller initially pays the notional of the trade.
2. Protection Seller receives premium payments on the current notional.
3. If no default occurs, Protection Seller receives entire notional at maturity.
4. If a default of occurs, two amounts are calculated:

a) the loss amount owed to the Protection Buyer.

b) the amount the notional of the trade is reduced.

The reduction in notional means the Protection Seller will not receive that amount at maturity, and thus counts toward the default payment the Protection Seller needs to make. This notional reduction can be greater than the loss amount actually owed, in which case the Protection Seller has overpaid and should obtain a rebate equal to the difference between the notional reduction and the loss amount.

12.2.1 NPV of Funded CDS Basket

To compute the cashflows, break the CDS lifetime into sampling periods. For the *ith* period, suppose loss L_i occurs with an associated drop in notional N_i ; then there is a net payment of $L_i - N_i$ which flows back to the Protection Seller. Summing over the probability of each loss amount per period and then discounting from sample payment date t_i^P produces the NPV of the default payments:

$$NPV_{creditPmts} = \sum_{i=1}^{n} D(t_i^{\ P})E[L_i - N_i]$$

Evaluating this expression will depend on the type of trade involved.

For a CDS Index Tranche or Nth Loss trade on an equity or mezzanine tranche, the notional drop of the tranche is equal to the loss amount assigned to the tranche. So in this case, $L_i = N_i$, and the NPV of the credit payments is zero.

For a CDS Index (whole basket trade), the loss amount if the *bth* asset defaults is $(1\text{-}R_b)n_b$, with asset notional n_b and recovery rate R_b. The notional drop of the basket is the full asset notional n_b, so the net payment, loss minus notional, is $- R_b\ n_b$, which is a rebate back to the Protection Seller. This is analogous to the single-asset CDS. Using one of the CDS Basket models, one can compute the expected loss for each sampling period and therefore the NPV of these payments.

In addition to the payments at default there are the principal payments. The payment at maturity will have an expected value determined by the expected loss amount over the life of the CDS. If the initial payment is in the past, what remains has value

$$NPV_{prin} = D(T)E[N(T)],$$

the discounted expected notional at maturity.

12.2.2 Breakeven Rate of Funded CDS Basket

The definition and formula for the Breakeven Rate of a basket trade is the same as that of a single-name CDS. That is:

$$b = s + \frac{\sum_{k=1}^{M} N_k - NPV}{Risky01}$$

12.2.3 Funded CDS Index Price from Quote

Assuming the market quotes spread for unfunded CDS Index trades, how does one find the price of a funded trade?

Rearranging the formula for the breakeven rate of a funded CDS, one has

$$NPV = \sum_{k=1}^{M} N_k - Risky01 * (b - s)$$

In pricing from a quote, the Spread DV01 is computed using a flat spread probability curve in the same manner as for the unfunded CDS, as the premium flows are the same in both cases.

Note that if the b = s, the NPV will just be the sum of the future principal, following the definition of breakeven rate for a funded trade.

Section 13. Senior Amortization and Incurred Recovery

13.1 Calculation of Incurred Recovery and Incurred Loss

When the detachment point of a tranche is very high, the recovered amount of defaults can reduce the protection afforded by the deal. The present section describes standard calculation of the effects of this "senior amortization".

Define:

N_0: Original Swap Notional

B_0: Implicit Notional of Entire Basket at time of origination

N(i): Outstanding Swap Notional after the ith default

a: Original attachment point of the tranche as a percent of B_0

d: Original detachment point of the tranche as a percent of B_0

In terms of monetary amounts, the original attachment amount A and detachment amount D are

$$A = aB_0$$
$$D = dB_0$$

So the Original Swap Notional is related to these by:

$$N_0 = D - A = (d-a)B_0 \,.$$

Note the notional B_0 of the basket is implicitly specified if one is given the tranche percentage size *d-a* and the Original Swap Notional.

As defaults occur, protection payments are made, the Outstanding Swap Notional *N(t)* decreases and the premium payments decrease accordingly.

When an asset defaults, its notional is divided into a Loss Amount and a Recovery Amount. The tranche defines a Loss Threshold and a Recovery Threshold. As defaults accumulate, the cumulative Loss Amounts can eventually cross the Loss Threshold, and the cumulative Recovery Amounts can eventually cross the Recovery Threshold. The crossing of the thresholds results in an Incurred Tranche Loss and Incurred Tranche Recovery, which decrease the size of the tranche. The Outstanding Swap Notional at any time is what remains of the tranche after the incurred loss and recovery have been removed.

The formulas for this procedure are as follows. Let there be a sequence of defaulting assets, labeled in order of default by i = 1, 2, 3.... Define

L_i: Loss amount of the ith asset to default

R_i: Recovery amount of the ith asset to default = Notional of ith defaulting asset minus L_i

The cumulative loss and recovery amounts for a sequence of defaults up to the ith default are formed by summing these for all prior defaults:

$$L_{Cum}(i) = \sum_{j<=i} L_j \;,$$

$$R_{Cum}(i) = \sum_{j<=i} R_j \;.$$

Note the cumulative amounts make no reference to the tranche levels.

Define the *Loss Threshold*, L_T, and *Recovery Threshold*, R_T, to be the amounts:

$$L_T = A\,,$$
$$R_T = B_o - D$$

with A and D the attachment and detachment amounts, as defined previously.

The cash settlement amount C_i for ith default is the amount of the loss L_i that contributes to the incurred tranche loss, up to the available remaining swap notional N(i-1) prior to the default. This can be written:

$$C_i = Min(N(i-1), L_i, Max(0, L_{Cum}(i) - L_T))$$

This will be zero unless the cumulative loss has crossed the threshold. In more detail: This Min function has three arguments. The last argument is the minimum when the loss threshold has not been crossed or has been crossed for the first time at the ith default; subsequent defaults will contribute the entire loss Li of the asset (the second argument of the Min); while defaults that exhaust the tranche will be limited in their effect by the tranche size prior to that default, N(i-1).

Similarly there is a recovery amount incurred by the tranche for each default in which the cumulative recovery has crossed the R_T threshold. Although this amount is not paid out in cash, it does draw down from the tranche notional. For the ith default, this incurred recovery I_i is

$$I_i = Min(N(i-1), R_i, Max(0, R_{Cum}(i) - R_T))$$

Summing up these quantities for all defaults in a sequence, one finds the total incurred loss payments and tranche recovery:

$$C_{Total}(i) = \sum_{j<=i} C_j\ ,$$
$$I_{Total}(i) = \sum_{j<=i} I_j\ .$$

The Outstanding Swap Notional after the ith default is defined to be the Original Swap Notional minus the incurred totals:

$$N(i) = N_0 - C_{Total}(i) - I_{Total}(i)$$

with the restriction that N(i) cannot fall less than zero. When the swap notional is zero, the deal terminates.

For the special case of protection on a whole basket, such as for a CDS Index swap, the tranche runs from 0% to 100%, and one can verify in this case that the Outstanding Swap Notional will always be the Original Swap Notional minus the sum of the notionals of each defaulted asset:

$$N(i) = N_0 - L_{Cum}(i) - R_{Cum}(i) = N_0 - \sum_{j<=i} n_i$$

where n_i, the ith asset notional, is L_i+R_i.

13.2 Example

As an example, consider a basket of just four assets of equal notional, on which a CDS protects 22% to 80% of the basket losses, with an Original Swap Notional of $58,000,000. The CDS is specified to use senior amortization. We wish to calculate the Outstanding Swap Notional after each asset defaults.

The original tranche quantities are

N_0 = $58,000,000

a = 22%

d = 80%

$B_0 = N_0 / (d - a)$ = $100,000,000

A = $22,000,000

D = $80,000,000

Because there are four assets in an implicit $100,000,000 basket, each asset has an implicit notional of $25,000,000.

One finds the loss and recovery threshold amounts:

L_T = A = 22,000,000

R_T = 100,000,000 - 80,000,000 = 20,000,000

Default Number 1:

Now let one asset default with a recovery of 60%.

(In the following, m stands for million.)

1st Loss Amount. L_1 = 25m * (1 - 0.6) = 10m

Cash Settlement . C_1 = Min(10m, Max(10m – 22m, 0), 58m) = 0. (Tranche not hit)

1st Recovery Amount. R_1 = 25m * 0.6 = 15m

1st Incurred Recovery. I_1 = Min(15m, Max(15m – 20m,0), 58m) = 0. (Recovery threshold not hit)

Outstanding Swap Notional = **Original Swap Notional** - **Total Incurred Loss Amount - Total Incurred Recovery Amount (not less than 0):**

N(1) = Max(58m – 0m – 0m, 0) = 58,000,000.

There has been no payment and no change in the swap notional.

Default Number 2:

Let second asset default with a recovery of 30%.

2nd Loss Amount. L_2 = 25m * (1 - .3) = 17.5m

Cumulative Loss Amount. $L_{Cum}(2)$ = 10m + 17.5m = 27.5m

Cash Settlement. C_2 = Min(17.5m, Max(27.5m – 22m, 0), 58m) = 5.5m. (Tranche hit)

2nd Recovery Amount. R_2 = 25m * .3 = 7.5m

Cumulative Recovery Amount. $R_{Cum}(2)$ = 15m + 7.5m = 22.5m

2nd Incurred Recovery. I_2 = Min(7.5m, Max(22.5m – 20m,0), 58m) = 2.5m. (Recovery hit)

Total Incurred Loss. $C_{TotaL}(2) = C_1 + C_2$ = 5.5m

Total Incurred Recovery. $I_{TotaL}(2) = I_1 + I_2$ = 2.5m

Outstanding Swap Notional. N(2) = 58m – 5.5m – 2.5m = 50m

There has been a cash protection payment of 5,500,000. The swap notional has been reduced to 50,000,000 because of the combined effect of the cash payment and the incurred recovery: 2,500,000 has been recovered and need no longer be covered by swap protection.

Default Number 3:

Let the third asset default with a recovery rate of 60%.

3rd Loss Amount. L_3 = 25m * (1 - .6) = 10m

Cumulative Loss Amount. $L_{Cum}(3)$ = 10m + 10m + 17.5m = 37.5m

Cash Settlement. C_3 = Min(10m, Max(37.5m – 22m, 0), 50m) = 10m

3rd Recovery Amount. R_3 = 25m * .6= 15m

Cumulative Recovery Amount. $R_{Cum}(3)$ = 15m + 15m + 7m = 37.5m

3rd Incurred Recovery. I_3 = Min(15m, Max(37.5m – 20m,0), 50m) = 15m

Total Incurred Loss. $C_{TotaL}(3) = C_1 + C_2 + C_3$ = 0 + 5.5m + 10m = 15.5m

Total Incurred Recovery. $I_{TotaL}(3) = I_1 + I_2 + I_3$ = 0 + 2.5m + 15m = 17.5m

Outstanding Swap Notional = 58m – 15.5m – 17.5m = 58m – 33m = 22m

There have been a total of two cash payments amounting to 15,500,000, and because of the recovered amount the swap notional has been reduced to 22,000,000. This does not quite cover the remaining single asset in the basket of notional 25,000,000 -- an effect of the original choice to limit protection to the 22-80% tranche.

Section 14. Random Recovery Model

14.1 Crisis and Recovery

In the credit crisis of 2008 the standard Gaussian Copula model ceased to function for senior index tranches. The problem is demonstrated in the following table. The maximum correlation possible in the model is 100%, but even at this maximum level the extremely high spreads for senior and super-senior tranches cannot be produced.

Tranche	Spread	Base Corr
0-3%	500.00	
	68.51%	39.5
3-7%	773.99	64.0
7-10%	435.52	73.3
10-15%	240.05	86.4
15-30%	126.50	-
30-100%	69.57	-

Market Quotes for CDX-NAIGS9V1-5Y for March 12 2008

The only other model parameters are the asset default probabilities and the recovery rates. A common interpretation of the breakdown among traders is that the market expects recovery rates at senior tranches to be lower than those expected of the individual assets. If defaults increase to the point where the senior tranches are hit, recovery rates would need to be downgraded to reflect a surprisingly poor economic environment. Thus what is relevant is the recovery rates *conditional* on a tranche being hit.

Trader perception is that recovery rates are negatively correlated with defaults -- that is, when the probability of default increases so does the anticipated loss amount given default. Capturing this negative correlation in a model is one way to vary the recovery rates of the Gaussian model. To do this, recovery rates must be made into random variables. A description of a general approach to random recovery is given by Andersen and Sidenius (2004).

Calypso has investigated the random recovery models that have been as well as internally developed models. For efficiency and smoothness, the best model has proved to be that published by Amraoui and Hitier (2008). Under general assumptions of parameters, calibration of base correlations is fast and the resulting deltas are smooth. The following table shows the results for the data given previously.

Tranche	Spread	Random Recovery Corr
0-3%	500.00	
	68.51%	39.45
3-7%	773.99	54.48
7-10%	435.52	60.71
10-15%	240.05	70.3
15-30%	126.50	87.83
30-100%	69.57	81.29

Random Recovery Correlations, CDX.NA.IG 9 5Y, March 12 2008

It should be understood that these correlations do not have the same interpretation as the original Gaussian model: that is, the same "correlation" applied to each model will produce different loss amounts and NPVs. The correlations that differ between these two tables do so in order to produce the same loss amount in the two different models, as they both must in order to fit the market quoted spreads. The deltas (sensitivity to quoted spread) or the tranches will differ between the two models, and the Random Recovery version is much closer to broker quotes in this case.

Details of the model can be found in the Calypso white paper, "Random Recovery Models for CDO Tranche Pricing".

14.2 RECOVERY_MODEL Parameter

Correlation Surface Generation: RECOVERY_MODEL Generator Parameter

The BaseCorrelation generator has a **RECOVERY_MODEL** parameter with two choices, **Fixed** and **Random**. The "Fixed" choice corresponds to the original Gaussian model, i.e., there is a fixed recovery rate assigned to each asset. Selecting "Random" will access the Random Recovery model.

Pricing: RECOVERY_MODEL Pricing Parameter

When pricing a trade using the one factor model, the default behavior is to use the same recovery model as on the surface. If the surface was generated with RECOVERY_MODEL set to Random then the CDSIndexTrancheOFM will use the Random Recovery model, and otherwise it will use the original method. On the pricer there is a Pricing Parameter that controls this behavior, named, like the generation parameter, **RECOVERY_MODEL.**

The choices are **As_Corr_Surface, Fixed**, and **Random**. The default is As_Corr_Surface, which has the behavior just described: the pricer model is kept in synch with the surface. But by selecting one of the other choices the user can force the choice of model regardless of the surface. This is useful for testing with manual input, for example.

References

Amraoui,S. and Hitier, S., "Optimal Stochastic Recovery for Base Correlation," BNP Paribas (June 2008)

Andersen,L. and Sidenius, J., "Extensions to the Gaussian Copula: Random Recovery and Random Factor Loadings", *Journal of Credit Risk* 1 1 (Winter 2004/05)

Calypso, Random Recovery Models for CDO Tranche Pricing (Calypso 2008)

Section 15. CDO Squares

A CDO Square is a CDO (CDSNthLoss) that pays out losses from a collection of component CDOs. It is represented as a CDSNthLoss, paying Par minus Recovery, whose underlying basket is in turn a collection of baskets with associated loss layers (tranches) defined for each. It is entered into Calypso by first defining a composite basket formed from basket tranches and using that as the reference basket of a CDSNthLoss trade.

CDO Squares are valued with Monte Carlo using `PricerCDSNthLoss`. This pricer will detect that the trade has the appropriate underlying for a CDO Square, and will direct the valuation to the appropriate model class, `MonteCarloCDOSquare`.

The default event payout of a CDO Square is defined as follows. Suppose the jth basket has suffered a cumulative loss of Cj,

$$C_j = \sum_{i \in D(j)} (1 - R_i) N_i$$

where the *ith* asset in the basket has recovery rate R_i notional N_i, and the sum runs over the assets *D(j)* in the *jth* basket that have defaulted.

From the cumulative loss one finds the loss *Lj* of the tranche of *jth* basket in the usual way for CDOs:

$$L_j = Min(E_j - A_j, Max(C_j - A_j, 0))$$

where A_j and E_j are the attachment and detachment points of the tranche of the *jth* basket (upper and lower bounds of the loss layer, expressed as monetary amounts).

Then the cumulative loss of the CDO Square is found by summing up the losses of each component tranche:

$$C = \sum_j L_j$$

The loss to the CDO Square's tranche is therefore

$$L = Min(E - A, Max(C - A, 0))$$

for attachment and detachment points *A* and *E* of the CDO Square's tranche. The termination payments are equal to the changes in L that occur at each default.

The Monte Carlo valuation of the CDO Square proceeds in the same fashion as a CDSNthLoss, with the change that the termination payments are computed as described above. The first step in the evaluation is to identify the unique assets in the component baskets, as the baskets may overlap in their contents. These unique assets are then used in the Monte Carlo sampling, with default times for the unique assets found for each future scenario.

These default times are used to compute the sequence of losses of the component baskets, and those losses used to find the loss and thus the termination payments of the CDO Square. The payments are discounted as usual. The process is repeated for the desired number of sampled scenarios and the results averaged to determine the expected present value of the termination payments.

The premium payments are computed for the scenarios just as in CDSNthLoss, with the difference that the CDO Square loss computation is used to adjust the notional of the premium after each default.

Section 16. Credit Default Swaptions

A Credit Default Swaption is an option on a Credit Default Swap (CDS). The strike price of a Credit Default Swaption is the CDS premium spread. A "right-to-receive" (RTR) swaption gives the purchaser an option to receive CDS premium (sell protection), and a "right-to-pay" (RTP) swaption the right to pay premium (buy protection). Other names for a Credit Default Swaption are Credit Spread Option and CDS option.

The following describes first the valuation of European swaptions on single-name credit default swaps. This is followed by approaches to pricing options on CDS indexes (baskets).

16.1 Expiration Payout of Single-Name Swaption

At expiration, the credit default swaption has a value equal to that of its underlying CDS, if that is positive (assuming the swaption holder would not exercise if the CDS NPV were negative). The CDS premium leg has a value determined by the strike spread of the swaption and the prevailing discount rates:

$$\text{NPV Premium} = w\sum_i Na_i D(t_E, t_i) P(t_E, t_i) K$$

with

K: The strike spread of the swaption, the spread of the underlying CDS

N: Notional amount of the CDS

a_i: The accrual year fraction of the ith premium payment

D (t, t_i): The risk-free discount factor at time t for payment date t_i

P (t, t_i): The probability at time t that the asset has not defaulted before the payment date t_i

t_E: Expiration time

w: Direction of premium:

- For RTP (pay premium; call on the spread: w = 1
- For RTR (receive premium; put on the spread): w = -1

This expression assumes the CDS does not pay partial accrual of premium in the case of default. Similar equations hold for other accrual types.

The opposite leg of the CDS, the credit default leg, has a value equal to the current market CDS spread, as the fair NPV of a market CDS is zero. So let:

S: The market spread of the CDS as of the expiration date.

Then the credit default leg has value

$$\text{NPV Credit} = -w\sum_i Na_i D(t_E, t_i) P(t_E, t_i) S$$

The total value of the swaption is then the sum of the two legs, if that value is positive:

$$\text{Swaption Expiry Value} = \sum_i Na_i D(t_E, t_i) P(t_E, t_i) Max(wK - wS, 0)$$

16.2 Valuation of Single-Name Swaptions

There is a standard method for pricing European credit default swaptions using a formula of the Black-Scholes type. This is described in the references listed below. In particular, Jamshidian and Hull & White derive the formula in a manner exactly analogous to that used for interest-rate swaptions, the only new feature being the presence of survival probabilities in the expected values of the cashflows.

This method is only applicable to swaptions with these features:

- European exercise
- Underlying CDS has a fixed premium leg (not floating)
- Underlying CDS is a single-name asset (not a basket)

Calypso implements this method in the class `PricerCreditDefaultSwaption`. The formula is described in the next section.

References

A. Arvanitis and J. Gregory, *Credit* (Risk Books, 2001) pp. 146-148

F. Jamshidian, "Valuation of Credit Default Swaps and Swaptions" (NIB Capital Bank, 2002)

J. Hull & A. White, "The Valuation of Credit Default Swap Options" (Univ. of Toronto, January 2003)

P. Schonbucher, "A Libor Market Model with Default Risk" (Bonn University, 1999)

16.3 Pricing Formula

A formula of the Black-Scholes type can be used to price a credit default swaption under the assumptions typical of the Black-Scholes environment:

- lognormal distribution of the forward rate, *S*
- constant volatility, σ
- fixed interest rate curves

as well as

- constant recovery rate
- fixed survival probability curves.

The logic behind the formula is exactly that of interest-rate swaptions, but the quantities in the formula are computed using CDS methods.

The forward rate S is the predicted breakeven rate of the underlying forward-starting credit default swap. The calculation of the breakeven rate is performed using `PricerCreditDefaultSwap`. The market data inputs to this calculation are the survival probability curve for the reference name, the assumed recovery rate, and the risk-free discount curve.

The volatility σ is usually obtained from market quotes. Calypso creates a volatility surface for CDS spreads from the issuer of the underlying CDS, and the volatility of the appropriate tenor is interpolated from this surface.

The discounting is performed using a combination of the risk-free discount curve and the probability curves, as described in the previous section on the payout value at expiration.

Then the formula for the present value of a credit default swaption for valuation date t_0 and time to expiry T is:

$$\text{SwaptionNPV} = w\sum_i Na_i D(t_0, t_i) P(t_0, t_i)[SN(wd_1) - KN(wd_2)],$$

$$d_1 = (\log(S/K) + \sigma^2 T/2)/\sigma\sqrt{T}$$

$$d_2 = d_1 - \sigma\sqrt{T}$$

This is the value for an underlying CDS does not pay accrued premium at default. For other types of CDS, the calculation of the underlying cashflows is done appropriately by the CDS pricer.

The "Knock-Out."

One question that this formula raises is whether it takes into account the "knock-out" property of a credit default swaption, something that does not exist in an interest-rate option. The knock-out occurs if the reference entity of the underlying CDS defaults prior to the option expiration date. In that case, the option immediately becomes worthless. However, the holder of the option may be compensated for this case.

Options on CDS Indexes have protection for the purchaser in case of default of any entities in the index basket. Valuing this pre-expiry protection requires a change to the formula. This is described in the section on CDS Indexes.

For single-name swaptions, there is almost always no default protection. Then the formula above is complete as it stands. Suppose the reference entity has defaulted prior to expiry, the pricing formula either should not be applied at all, or it should be used with the corrected probabilities: that is, all survival probabilities need to be set to 0.0. Then the NPV of the swaption is immediately zero.

But if the reference entity has not defaulted, the knock-out is nonetheless implicitly taken into account through the survival probabilities $P(t_0, t_i)$. These are the probabilities *at the value date t_0* that the entity survives to t_i. In particular, at the expiry date t_E, which is the start date of the underlying CDS, one would have $P(t_0, t_E) < 100\%$. The possibility that default of the underlying entity has occurred by the expiry date is therefore already taken into account by using the current survival probabilities. There is no need to apply a separate knock-out logic.

16.4 Required Market Data

16.4.1 CDS Volatility Surfaces

The pricing formula requires the volatility of CDS spreads as in input. This is provided to the pricer by a volatility surface that is associated with those spreads. The user needs to set up this surface before pricing, or else manually supply a transient volatility for pricing in the trade window of the Credit Default Swaption.

To set up a volatility surface:

1. Create a surface using Main>Market Data>Surfaces>VolSurface3D. For the "Vol Type" field select "Credit". Otherwise the surface is made similar to an interest-rate swaption surface.

2. In the PricerConfig, select the Credit tab.

3. Specify a Currency, Issuer and Seniority. Set the "Market Data Usage" field to "VOL". Select the credit volatility surface. Add the item to the list.

When the volatility is looked up in the `PricerCreditDefaultSwaption`, the surface is looked up by Currency (a String), Issuer ID (an integer), and Seniority (a String):

```
pricerConfig.getCreditMarketDataItem(PricerConfig.VOLATILITY,
aCurrency_String,
anIssuerID_integer,
aSeniority_String).
```

16.4.2CDS Market Data

The other market data required for CD swaptions is used for valuing the underlying CDS. This includes discount curves and a survival probability curve. If the data has already been set up so that the CDS trade can be valued on its own then that same data will be used by the swaption.

Section 17. CDS Index Options

17.1 Generalization for Index Basket Options

A CDS Index is a standardized basket of reference entities. Quotes on spreads for credit default swaps using a CDS Index as an underlyer are usually readily available in the market.

Options on CDS Index swaps are valued in Calypso by first creating an equivalent Credit Default Swaption whose underlying is a Credit Default Swap on the basket defined by the index. Then that swaption is evaluated using a method analogous to that described for single-name credit default swaptions.

In order for the formula of the Black-Scholes type to be applicable to baskets, some assumptions need to be made.

- The forward spread of the CDS Index is assumed to be lognormally distributed.
- The volatility of the CDS Index spread is assumed to be obtained from the market quotes for the index swaps.

Under these assumptions, the formula can be used as long as the forward rate and the normalization factor are computed appropriately for the CDS Index swap.

The forward rate *S* is found using the `PricerCDSIndex` to solve for the breakeven rate for the forward-starting CDS index swap. The computation finds the NPV derived from the probability curves of all the underlying names, rather than from a market quote for the CDS Index; that is, the parameter NPV_FROM_QUOTE is set to False.

The normalization factor is found by finding the NPV of the premium leg of the underlying index swap and dividing by the option strike spread. Note this involves all the probability curves of the underlying reference entities. This NPV is found as of the value date t_0 for the leg starting at delivery date t_{del}, so write the normalization factor as:

$$V(t_0, t_{del}) = \frac{\left|\text{NPV Premium}(t_0, t_{del})\right|}{K}$$

Then the revised formula is, if there is no default protection,

$$\text{IndexOptionNPV} = w\, V(t_0, t_{del})\,[SN(wd_1) - KN(wd_2)].$$

17.2 Cash Settlement of CDS Index Options

CDS Index options that are cash settled will exchange a fee at expiration equal to the fee paid on a market CDS Index trade. The market convention for computing this fee is described in the documentation on the pricing of CDS Index trades.

Note: At this writing, Calypso only supports physical exercise of CDS Index options. Cash settlement is forthcoming.

17.3 Pre-Expiry Default Protection

Unlike single-name credit default swaptions, options on CDS indexes usually trade with a guarantee of protection against default of names in the basket that occur prior to expiry. At exercise, if there were any losses to the basket due to defaults during the option holding period, they are summed up and included in an upfront fee paid to the premium payer, and the notional of the basket is reduced by the defaulting principle (the full principle, not just the losses).

The swaption formula can be modified to take into account the pre-expiry default protection feature. The following is used in the marketplace and is implemented by Calypso.

Because the purchaser an the option to pay premium will receive the payments from losses on the basket, the value received at expiry is estimated as the present value of the expected losses from the CDS index option as of the current date. The underlying CDS index trade starts after expiry, but the losses start up now, so the option can be considered as paying out a protection leg of an extended CDS index trade that starts at the trade date of the option and matures at the maturity date of the underlying CDS index trade.

The value of this protection leg can be written as of the current date t_0 as:

$$NPV_{protection,extended} = V(t_0, t_{spot})S$$

where $V(t_{spot})$ is the *Spread DV01*, also known as the *risky duration* or *risky PV01*, of the extended CDS index trade, valued as of the spot date relative to the trade date. *S* is the spot rate of the CDS index on that date. (Please see the documentation on CDS Index valuation for more information on Spread DV01.)

At exercise, the payer of premium will be committed to pay the NPV of the premium leg of the underlying CDS index that starts after the expiry date, paying the strike premium K. Write its value as of the current date t_0 as

$$NPV_{premium} = V(t_0, t_{del})K.$$

Here the Spread DV01 is computed as of the delivery date t_{del}.

The payoff value of the RTP swaption is then

$$P = Max(V(t_0, t_{spot})S - V(t_0, t_{del})K, 0).$$

This can be rewritten in terms of an *adjusted spot rate,* S_{adj} and adjusted strike K_{adj}

$$\begin{aligned} P &= V(t_0, t_{del})Max(\frac{V(t_0, t_{spot})S}{V(t_0, t_{del})} - K_{adj}, 0) \\ &= V(t_0, t_{del})Max(S_{adj} - K_{adj}, 0), \end{aligned}$$

where

$$S_{adj} \equiv \frac{V(t_0, t_{spot})S}{V(t_0, t_{del})}.$$

and

$$K_{adj} \equiv C + (K - C)\frac{V(t_0, t_{spot})S}{V(t_0, t_{del}) * Q(t_0, t_{del})}.$$

where C is the index coupon and Q is the survival probability.

One now makes the assumption that S_{adj} is lognormal. In that case, the swaption formula is of the Black-Scholes type. The result is:

$$\text{CDSIndexSwaptionNPV} = wV(t_0, t_{del})[S_{adj}N(wd_1) - K_{adj}N(wd_2)],$$
$$d_1 = (\log(S_{adj}/K) + \sigma^2 T/2)/\sigma\sqrt{T}$$
$$d_2 = d_1 - \sigma\sqrt{T}$$

This is the simple model for a CDS Index swaption with pre-expiry default protection.

17.4 Arbitrage Free Credit Index Option Valuation

This section provides details of the arbitrage free credit index option pricing.

The standard Black's approach for pricing index options is to add front end protection: FEP =

$$(1 - R)\, D(t, t_E)\big(1 - Q(t, t_E)\big)$$

to the standard black's formula to account for the protection from the option initiation to exercise date.

$$V^{pay}(t) = (1 - R)\, Z(t, t_E)\big(1 - Q(t, t_E)\big) + RPV01(t, t_E, T) * (F\, \Phi(d_1) - K\, \Phi(d_2))$$

Where R is the recovery, Q is the survival, RPV01 is the risky present value, F forward spread and K strike spread. However, as observed by Pedersen1, this model does not capture the exercise decision properly. This can be seen for the case of payer credit index of swaption in the limit strike K -> ∞ option should never be exercised and hence the value of the option should go to zero. However, from the above formula above in the limit K -> ∞ option value tend to the value of the front end protection. Further the black's method does not satisfy the following put-call parity condition when the strike spread (K) differs from index coupon C(T).

$$V^{pay}(t) - V^{rec}(t) = \big(S_I(t, t_E, T) - C(T)\big)\, RPV01(t, t_E, T) - \frac{\big(K - C(T)\big) RPV01(t, t_E, T, K)}{Q(t, t_E)} + FEP(t, t_E)$$

In the sense that it black's model does not satisfy put-call parity, it is not arbitrage free. To improve upon black's model, approach based on Andersen's2 /O'Kane3 has been implemented. This approach is arbitrage free and satisfies put-call parity. In the following, arbitrage free approach has been discussed in further detail.

17.4.1 Option PayOff

Payoff Consists of payments/receipts at three different times: Option contract initiation time t=0, option expiry time t_E and during the index maturity time (t_E,T). These payments for an payer index option (i.e., exercising is equivalent to entering into a contract for buying protection) are

– Payment of option premium on option initiation time t=0.

---

[1] Pedersen, C.M., *"Valuation of portfolio credit default swaptions". Technical report, Lehman Brothers, New York, 2003.*

[2] Andersen, L., *"CDS Options, CMDS, general PDEs and index options"(Lecture Notes 2006)*

[3] Dominic O'Kane., " Modeling single-name and multi-name credit derivatives", Wiley-Finance, 2008.

- On t_E payoff consists of two payments, provided option is exercised, which depends on the present value of the index as of exercise date. One to receive the money from option seller NH_0,

$$\mathbb{E}(H_0) = \frac{1}{M} * \sum_{m=1}^{M} (1 - R_m)(1 - Q_m\,(0, t_E) \;\approx (1 - R)(1 - Q(0, t_E))$$

  for the defaulted credits between option initiation time and exercise date accounting for recovery R_m.

  Another is the payment of exercise price

$$G(K) = \; N\big(K - C(T)\big)\, RPV01_I(t_E, T, K)$$

  to the option seller , where S_{index} index spread, *RPV01* is the risky present value , *C(T)* is index coupon. This is the value of the CDX (on full original notional before any defaults, since defaulted notional is being paid) priced using an index spread equal to strike K.

- Option buyer receives the CDX index from the option seller, whose present value is given by the current market index spread S_{index}, providing protection on the rest of the non-defaulted credit till index maturity(T). The value of this long protection on exercise date is given by

$$N\,\big(S_{index}\,(t_E, T) - C(T)\big) \; * \; RPV01_I(t_E, T, S_{index}\,(t_E, T))$$

  where the *RPV01*, at time t_E is calculated using a flat credit curve using the spread S_{index} on the expiration date.

  Where

$$RPV01_I(t_E, T, S_{index}\,) = \sum_{n=1}^{N_T} a_n\; D(t_E, t_n)\;(\mathrm{Q_n} + \;0.5 * (\mathrm{Q_{n-1}} - \mathrm{Q_n})\,(1 + \;\delta_{k1} * \;f))$$

  where f is the fraction of the first period accrued so far.

Total payoff then would be

$$w * N * \big(H - G(K)\big)$$

where

$$H \; = \mathrm{H_0} + \big(S_{index}\,(t_E, T) - C(T)\big) \; * \; RPV01_I\big(t_E, T, S_{index}\,(t_E, T)\big)$$

Where *w* is the direction of the premium (assumes values of +1 for RTP option and -1 for RTR option). The holder of the option will exercise only if the payoff is greater than zero. As a result, we have payoff on exercise date t_E

$$Payoff(t_E) = \max[w * N * \big(H - G(K)\big), 0]$$

which provides option value *CDSIndexSwaptionNPV(t)*

$$CDSIndexSwaptionNPV(t) = \; N\, D(t, t_E)\; \mathbb{E}[\max(H\big(t_E, S_{index}\,(t_E, T)\big) - G(K)\,, 0)]$$

Provided there have been no defaults before option expiry, a payoff of zero occurs when

$$S_{index}\,(t_E, T) = K$$

However when there are defaults before option expiry date t_E, option could be exercised even when the spread is less than option strike, due to the payment of protection for the defaulted credits before the expiry date t_E.

17.4.2Option Valuation

Since, the index spread S_{index} has a flat term structure, credit triangle approximation is justified and risky present values can be given as

$$RPV01_I(t_E, T, S_{index}\,) = \sum_{n=1}^{N_T} a_n D(t_E, t_n)\,\exp(-S_{index}\,(t_n - t_E)/(1-R)\,)$$

Where N_T is the number of the coupon payment dates between option expiry date(t_E) and index maturity date (T). Then the option value is given in the usual manner assuming S_{index} to be log-normal

$$S_{index}\,(X, Z) = X \exp(-\frac{\sigma^2 (t_E - t)}{2} + \sigma\, Z \sqrt{(t_E - \ t)})$$

where X is the normalizing factor. When using basis adjusted spreads expected value of index should be same as the sum of the expectation of the underlying credits

$$\int_{-\infty}^{\infty} H(S_{index}\,(X,Z))\,\phi(Z)\,dZ = H_0 + \frac{1}{M\,D(t,t_E)} \sum_{m=1}^{M} \big(S_m(t,t_E,T) - C(T)\big)\,RPV01_m(t,t_E,T)$$

$$H\big(S_{index}(X,Z)\big) = \ \big(S - C(T)\big) * \ RPV01_I(t_E, T, S_{index}\,(X,Z))$$

which can be rewritten from the definition of the S_{index}

$$\int_{-\infty}^{\infty} H(S_{index}\,(X,Z))\,\phi(Z)\,dZ = H_0 + \frac{1}{D(t,t_E)} \big(S_{index}\ \ (t,t_E,T) - C(T)\big) * RPV01_I(t,t_E,T,S_{index}\,)$$

Where

$$RPV01_I(t,t_E,T,S_{index}\,(t,t_E,T)) = RPV01_I(t,T,S_{index}\,(t,t_E,T)) - RPV01(t,t_E,S_{index}\,(t,t_E,T))$$

$$S_{index}\,(t,t_E,T) = (\frac{1}{M}\sum_{m=1}^{M} S_m(t,T) * RPV01_m(t,T,S_m) - \frac{1}{M}\sum_{m=1}^{M} S_m(t,t_E) * RPV01_m(t,t_E,S_m))\,/\,(\frac{1}{M}\sum_{m=1}^{M} RPV01_m(t,T,S_m) - \frac{1}{M}\sum_{m=1}^{M} RPV01_m(t,t_E,S_m))$$

The integration above is carried out numerically using Gauss-Legendre integration between limits of -4.0 to 4.0. The resulting equation can then be used to solve for X using modified brent iterative solver. Given X, lognormal distribution for S_{index} is fully specified, which can then be used for calculating option value

$$CDSIndexSwaptionNPV(t) = N\, D(t, t_E) \int_{-\infty}^{\infty} \max(wH(S_{index}(X,Z)) - w\, G(K), 0)\phi(Z)\; dZ$$

which again is evaluated using Gauss-Legendre numerical integration.

17.4.3Implementation

This is implemented as another approach of credit index option valuation in the class PricerCreditIndexOption when the pricing parameter USE_PEDERSEN_MODEL is set to true.

Section 18. Constant Maturity CDS (CMCDS)

A CMCDS is a floating-rate credit default swap whose premium rate is periodically reset to the market spread of a constant-maturity CDS.

A CMCDS is typically quoted in terms of a *gearing factor* that multiples the constant-maturity rate. In Calypso the user enters this as an *index factor* when defining the floating rate of the trade. In addition, the CMCDS could have an additional fixed "spread" or basis added to this result. (Note the dual use of the word "spread.")

18.1 CMCDS Rate Index

A Rate Index needs to be defined for the CMCDS in order to specify the characteristics of the constant-maturity swap, including its underlying asset – the latter will be referred to as the *reset asset* of the CMCDS index, to distinguish it from the *reference asset* on which protection is sold in the swap.

There is a default Rate Index simply named "CMCDS" which defines the reset asset to be the same as the reference asset of the CMCDS trade. It also defines other characteristics of the underlying constant-maturity swap to have the same values as the CMCDS trade, such as the frequency and day count of the premium payments.

If the CMCDS Rate Index is chosen, PricerCMCDS must be used in order to properly project the forward rates of this index.

18.2 CMCDS Pricer

The CMCDS pricer, class `PricerCMCDS`, will project the cashflows for a trade based on a CMCDS index and then use these in the standard way to compute NPV and sensitivities of the CDS.

To project the cashflows of a CMCDS trade, the following steps are performed in the pricer.

1. For each reset date, construct the underlying fixed-rate constant-maturity swap with the appropriate asset, time to maturity, payment frequency and daycount. Each reset date will correspond to the first reset date of one of these forward swaps. If the default Rate Index named "CMCDS" is used, the constant-maturity swaps created in this way will have the same reference asset, frequency, etc., as the CMCDS trade that is being priced.

2. Project the forward breakeven rate of each forward swap using the probability curve and discount curve of the reset asset.

3. Compute a convexity adjustment to the forward breakeven rate.

4. Form the net projected forward rate for the cashflow:

$$r_i = g * (F_i + c_i) + b$$

where

r_i: net projected forward rate of ith cashflow

g: gearing factor (index factor) defined on the trade

F_i: unadjusted breakeven spread of the ith forward constant-maturity swap

c_i: convexity adjustment to the ith forward breakeven spread

b: the spread-to-index (basis) defined on the trade

5. Complete the calculation of the premium cashflow by multiplying by the notional of the CMCDS and the accrual period of the payment.

The NPV is then computed as for any single-name CDS using these premium cashflows. The protection leg depends only on the reference asset and so its valuation is independent of the CMCDS index.

In the cashflow projection, the only new complexity is the calculation of the convexity adjustment. This is described in the next section.

18.3 Convexity Adjustment

The convexity adjustment for a constant-maturity swap forward rate is computed using the following approximation:

$$\text{Convexity Adj} = \frac{F^2}{1-R} \frac{\sum_{j=1} t_j D_j P_j}{\sum_{j=1} D_j P_j} (\exp(\sigma^2 T) - 1),$$

F: Unadjusted breakeven spread of the forward constant-maturity swap

R: Recovery rate assumption

t_j: Time from the start date of the forward swap to its *jth* payment date of the forward swap

D_j: The discount factor from the *jth* payment date of the forward swap back to its forward start date

P_j: Probability of survival of the reset asset to time t_j

σ: The volatility of the forward swap rate

T: Time from value date to payment date of the CMCDS cashflow that is reset from this forward swap breakeven rate

References:

"Convexity Conundrums: Pricing CMS Swaps, Caps and Floors", Patrick Hagan, Wilmott Magazine (March 2003)

"Constant Maturity CDS (CMCDS) – A Guide", Nomura Fixed Income Research (May 2005)

Section 19. CDS on ABS/ABX

19.1 Pricers

Credit default swaps on the price of an Asset-Backed Security (ABS) or an Asset-Backed Security Index (ABX) have the following Calypso pricers:

– Credit Default Swap on ABS - PricerCreditDefaultSwapABS

– CDS on ABS Index - PricerCDSABSIndex

PricerCDSABSIndex is a simple pricer that values a CDSABSIndex from a given quote as though it were a single-name CDS. Information from the pool underlying the ABS is not needed. Thus it is assumed the market quote captures information about the pool default payments.

19.2 Valuation using Market Quotes for Spread and WAL

ABS

To price a CDS on ABS from a quote for a breakeven spread, the method used is based on the market convention formula for pricing an ordinary CDS Index from a spread quote.

The formula is modified for ABS by applying a quote for the weighted-average life (WAL). The WAL is a quote associated with the ABS bond underlying the CDS. (The user can find this in the Quote Set by performing a search on ".WAL".)

This WAL is used to create an effective maturity date of an equivalent single-name CDS, by cutting off premium payments at the WAL time. From that point, the conventional valuation is performed. In brief:

- A vanilla CDS is made the same premium cashflow dates as the CDS on ABS truncated at the WAL, with premium rate equal to the quoted spread.
- The probability curve is a flat-spread curve, created by taking the quoted spread as the breakeven spread starting on the settlement date (spot date with respect to the value date).
- The recovery rate is 40%.
- Accrued premium is handled using market approximations.

(Please see this document's section on CDS Index valuation from a spread quote for further details.)

This is implemented in PricerCreditDefaultSwapABS. This pricer will look up the most recent WAL quote for the underlying ABS and does not require a daily update of that quote. As long as there is at least one past WAL quote, the pricer will not show the quote as missing when the user performs market data check.

ABX

An CDS on an ABX index can be quoted either as a clean price or as a premium spread.

If quoted as a spread, the NPV and the credit PV01 are computed as described in the section on standard CDS Index valuation.

If quoted as a clean price, the valuation is performed as described for a standard CDS price-based Index. That is, the NPV of is given by

$$NPV = wN(1-p) + wA$$

where

N: Notional of trade (a positive number)

A: Accrued premium of current period (a positive number)

p: Clean price per 1.0 of notional (usually a number near 1.0)

w = +1 if paying the premium, 1 if receiving the premium

Section 20. Credit-Linked Contracts

A *credit-linked* or *credit-contingent* contract is any contract, such as a bond or interest-rate swap, which is specified to terminate in part or in full when credit events (defaults, etc.) occur for one or more reference entities.

20.1 Credit-Linked Notes (CLNs)

20.1.1 Definition

A *credit-linked note (CLN)* is a funded interest-rate note with payments contingent on the credit worthiness of an entity other than the issuer. Thus the note bears additional credit risk to reference assets which the issuer transfers to the note investor. The investor is compensated for the additional risk by an enhanced coupon rate.

Payments to the investor can be partially protected to decrease the credit risk (and hence the coupon). Types of CLN's include the following:

- **No protection** -- When a credit event occurs, both coupons and principal are affected. At maturity (or upon default, if so specified) the principal is not returned. Instead, the investor receives the recovery value of the defaulted reference asset (or the reference asset itself, if physically settled).
- **Principal Protected** – When a credit event occurs, coupons are affected but principal is unaffected. Investor in the note receives full principal back.
- **Coupon Protected** – When a credit event occurs, the coupons continue to pay in full and on schedule. The principal payment disappears and is replaced by the recovery value of the reference asset (or the asset itself).

If the CLN depends upon a basket of reference entities, then each default will decrease the remaining principal, affecting the subsequent coupon or principal payments if they are not protected. If unprotected, the outstanding principal and/or coupon notional are:

- for an Nth Default, the remaining undefaulted notional of the basket;
- for an Nth Loss, the remaining tranche notional.

At maturity, the remaining principal is returned to the investor.

20.1.2 Calypso CLN Bond and Pricer Configuration

To define exactly how CLN payments are adjusted for credit events, the contract must specify the same information as in a Credit Default Swap on the reference entity or basket of reference entities. If the CLN is dependent on a reference basket it may be either an Nth Default or Nth Loss type, and the appropriate parameters must be specified in each case: the basket size and the number of defaults or the tranche attachment and detachment points. The types of credit events are also specified.

In the Calypso these are detailed in the Bond configuration window, with the bond class specified as BondCLN and the bond type either "Standard", if referring to a single reference asset, or "Basket" if referring to a basket of assets.

Calypso provides two pricers, PricerBondCLN and PricerBondCLNBasket, for the single and basket reference entity cases, respectively.

For single reference entities the former pricer will be used by default.

To use the basket pricer with a BondCLN trade, the Pricer Configuration "Pricers" tab **must** associate PricerBondCLNBasket with a BondCLN product with subtype "Basket".

Product	Subtype	Required Pricer Config Association
BondCLN	Standard	PricerBondCLN
BondCLN	Basket	PricerBondCLNBasket

20.1.3Valuation by CDS Replication

The valuation of CLN trades uses either bond or CDS pricing, depending on the parameter choices.

Quote-based pricing. If the CLN is priced from a quote, then the usual bond pricing is used. The quote is assumed to include the reference entity credit risk, so this valuation employs no additional computations of the credit-contingent contribution.

CDS-replication pricing. The CLN can be priced using the discount and forecast curves for the bond flows together with the survival probability curves for the reference assets. In this case, a Credit Default Swap is created to replicate the CLN, then the appropriate pricer for the Credit Default Swap is used. The NPV of the CLN is equal to the NPV of the replicating CDS.

For a default bond, when the issuer defaults, the standard convention is that there is no accrual coupon payment and only the recovery amount is received at maturity.

The formula to price the default bond is as follows:

$$NPV = \sum_{i=1}^{n} Coupon_i * DF_i * q_i + N * DF_n * q_n + N * DF_n * (1 - q_n) * (1 - R)$$

Where

$Coupon_i$ is the i-th coupon payment,

DF_i is the end date discount factor in the i-th coupon period,

q_i is the end date survival probability in the i-th coupon period,

N is the default bond notional,

n is the number of remaining coupon payments,

R is the recovery rate of the default bond.

The replicating CDS depends that is created depends upon the reference assets of the CLN:

If CLN type is...	CDS type is...
Single-Name CLN	CreditDefaultSwap
Nth Default Basket CLN	NthDefault
Nth Loss Basket CLN	NthLoss

The protection seller of the CDS is the investor (buyer) of the CLN. This is the party that receives the premium flows.

The reference asset(s) of the CLN are made the reference asset(s) of the CDS.

Premium Flows on the CDS are set equal to the contractual coupon and principal flows on the CLN.

The payment type on the CDS premium leg is set as follows:

If CLN coupon is…	CDS Premium Payment type is set to…
Coupon protected	RISKLESS
Coupon not protected	NO ACCRUAL

Note that if coupon isn't protected, premium type is assume to be NO ACCRUAL. That is, when a default occurs in the middle of a coupon period, none of the unprotected coupon gets paid.

The **CDS termination payment** is set as follows:

If CLN principal is…	CDS Credit Default Payment type is set to…
Principal protected	NONE
Principal not protected	PAR MINUS RECOVERY

If the principal is protected, in the case of default it is paid without a countervening protection payment. If it is not protected, then in case of default there is a PAR MINUS RECOVERY payment and the CDS is defined as FUNDED, which results in a net payment at maturity of the recovery amount rather than of the full principal.

The NPV of the replicating CDS is computed using the appropriate pricer for that CDS.

20.1.4CLN Issuer Risk

The NPV calculation described above currently does not include issuer risk, as the CDS valuation does not currently support counterparty risk. However, issuer risk can be estimated using the Jump to Default analysis.

For a CLN, the *Jump to Default analysis* will simulate a default in every reference asset in turn and compute the resulting post-default NPV and default exposure to that asset. Additionally, the issuer default is simulated by assuming a post-issuer-default NPV equal to the recovery value of the CLN. A recovery rate must be defined for the issuer (e.g., in an issuer probability curve) in order to perform this calculation.

In the resulting Jump to Default analysis report, the user will see a line for each reference entity, and one additional line for the issuer. If the issuer itself is one of the reference entities in the basket, there will be only one line showing that issuer, with post-default NPV being the CLN recovery value.

20.2 Credit Contingent Interest-Rate Swap

20.2.1 Definition

A credit-contingent interest-rate swap is a conventional interest-rate swap which terminates (or decreases in notional value) when a credit event occurs to a third-party reference entity. It is the swap analog of the credit-linked note.

The simplest type depends on a single reference entity. If the entity suffers a credit event, both legs of the swap are terminated. There are no more payments. However, the credit-contingent conditions may specify that a separate payment be made by one party or the other when a credit event occurs.

Similarly, a First-to-Default credit contingent swap will terminate at the first default in a basket of reference entities. A separate termination payment may be made at that time.

The credit-contingent conditions are similar to those found on a credit default swap. There are also some additional choices:

- *Contingency window*. The swap may include a start and an end date for the period in which there is a dependence on the reference entity. If the entity has a credit event before or after this period, there is no effect on the swap.
- *Termination payout*. If the swap terminates due to a credit event, one swap party may be required to make one of the following payments to the counterparty:

  PAR (i.e., the swap notional at the time of default)

  PAR MINUS RECOVERY

  FIXED AMOUNT

  FIXED PERCENTAGE of notional

  RECOVERY

  NONE (No payment)

20.2.2 Valuation by CDS Replication

Each leg of a credit-contingent interest rate (IR) swap can be regarded as the premium payment of a credit default swap, with the payments terminating (or decreasing in notional) when a default occurs in a reference asset. Thus the value of the IR swap should equal the value of two credit default swaps, if the protection payments are chosen to match that defined on the IR swap.

At least one of the replicating credit default swaps will have no protection payments. The direction of the protection payment on the IR swap determines which of the replicating credit default swaps will have that payment, which will be the swap with the premium payments in the opposite direction to that protection payment. For example, if the IR swap pays fixed coupons, receives floating coupons, and pays out a fixed amount $100,000 when a default occurs, then the replicating credit default swaps are:

1. Pay fixed coupon, no protection payment (pays fixed amount of 0.0)
2. Receive floating coupon, protection payment pays fixed amount $100,000 upon default

If there is a credit-contingency window defined, these will be applied to the replicating swaps, even though ordinarily credit default swaps do not employ such windows. All swaps are cash settled and assumed to be the “No Accrual” type of premium (no interest payment if default occurs within coupon period).

20.2.3 Credit-Contingent Single Swap Legs

A Single Swap Leg can be traded on its own, or used as an element in a structured product. Valuation of a credit-contingent single swap leg is done by replicating with a credit default swap as described above.

The only practical differences between a trade on a Single Swap Leg and on a credit default swap are the greater variety of choices on a Single Swap Leg: the contingency window and the variety and choice of direction of the terminating protection payment. Unlike the credit default swap, the termination payment can be in the same direction as the "premium" payments.

20.2.4Pricers

There are separate Calypso pricers for credit-contingent swaps and swap legs, and for single versus basket reference entities. These pricers perform the replication described above and computes the values using the configured credit default swap pricers. The following table demonstrates how the Pricer Configuration should be set up to use these pricers.

Product	Extended Type	Required Pricer Config Association
Swap	CreditContingent	PricerSwapCreditContingent
Swap	CreditContingentBasket	PricerSwapCreditContBasket
SingleSwapLeg	CreditContingent	PricerSingleSwapLegCreditContingent
SingleSwapLeg	CreditContingentBasket	PricerSingleSwapLegCreditContBasket

Section 21. Performance Swap

A *Performance Swap* pays the cashflows and periodic market price changes of a reference obligation in exchange for a regular fixed or floating rate funding. It is useful as a means of transferring credit risk, as the party paying the price changes will receive a net payment in the case of a credit event.

21.1 Valuation

To value a Performance Swap, each cashflow type is priced separately:

$$NPV = PV(Asset\ Coupons) + PV(Price\ Change\ Flows) + PV(Funding\ Flows)$$

The values of the premium leg and the cashflows from the reference asset are found in the conventional way for swap and fixed income cashflows, with credit contingency taken into account. The novel cashflow type is *the price-change cashflow* that pays out the market price changes.

A price-change cashflow is valued in two steps. First, the necessary future prices of the reference asset are projected, from which the amount of the cashflow is computed. Second, the present expected value of the cashflow is found using a riskless discount curve and a credit event probability curve.

21.1.1 Forward Price Projection

To calculate the amount of a price-change cashflow, the market prices on the start and end dates of the cashflow must be known or projected. A future market price is projected by using the pricer for the reference asset and pricing a hypothetical trade that settles on the future date. The specific method of finding a forward price can vary with the pricer; in Calypso's default pricers for bonds and loans, forward prices can be projected with or without a repo curve.

21.1.2 Expected Value Computation

The expected value computation is performed by finding the value of the future price-change cashflow under various credit outcomes and multiplying by the probability of each outcome. Define:

p_1: Probability of survival to the start date of the cashflow

p_2: Probability of survival to the end date of the cashflow

M_1: Market value of the reference asset at the start date of the cashflow (known or projected)

M_2: Market value of the reference asset at the end date of the cashflow (known or projected)

R: Recovery value of the reference asset upon default (assumed)

D: Discount factor on payment date of cashflow

If the reference asset has not yet suffered a default or other credit event, then there are three future outcomes.

1) Credit event occurring prior to cashflow end date:

Cashflow Amount Due = $M_2 - M_1$

Probability of survival to end date = p_2

2) Credit event occurring before start of period:

Cashflow Amount Due = 0

Probability of default before start = 1 - p_1

3) Credit event occurring between start and end of period:

In this case, the market value M_1 at the start of the period is known from a price reset, while the market value at the end of the period is assumed to equal the recovery value, which is calculated as the assumed recovery rate times the notional of the reference asset.

Cashflow Amount Due = R - M_1

Probability of credit event within the cashflow period = p2 - p1.

The expected value is the sum of the values for each outcome times the probability of that outcome. The present value found by multiplying the result by the riskless discount factor at the payment date.

$$Cashflow\ NPV = [p_2(M_2 - M_1) + (p_2 - p_1)(R - M_1)]D$$

Each future price-change cashflow is found in this way, and the result summed to provide the total price-change NPV.

21.2 Pricers

21.2.1 PricerPerformanceSwap

The calculations described in the previous sections are for the pricer PricerPerformanceSwap.

Log category for pricing details is “PerformanceSwap”.

21.2.2 PricerPerformanceSwapAccrual

In addition, Calypso also provides another pricer called PricerPerformanceSwapAccrual, to value performance swap products. The difference between PricerPerformanceSwapAccrual and PricerPerformanceSwap pricers lies in how they calculate the present value of the asset coupons and the funding flows, while the same for the price change flows. In PricerPerformanceSwapAccrual, it only includes the asset coupon accrual amount and the funding flow accrual amount computed up to valuation date. Thus, when using PricerPerformanceSwapAccrual, the value of performance swap becomes:

$$NPV = Asset\ Coupon\ Accrual + PV(Price\ Change\ Flows) + Funding\ Flow\ Accrual$$

Appendix

Log Categories

The following logging categories can be used to view internal details of calculations.

CDSModel Displays details of the NPV calculation for both legs of a CDS basket trade.

LossModel Displays detailed information about the loss distribution calculated by loss distribution models. The output of the loss distribution is the input to the NPV calculation logged in the "CDSModel" category.

PVCDSQuote Displays details of the calculation for pricing a CDS Index of Credit Default Swap from a market quote.

CorrGenReport Displays details of correlation surface generation. For even finer detail, use also with the "Solver" category to see how each correlation is found. (There is now a checkbox on the generation UI which performs this logging and displays the result in a popup window, without the need for the user to define the log category.)