Adenza

Calypso Bond Analytics

Version 17.0

February 2022 – First Edition

The purpose of the Bond Analytics guide is to provide an understanding of the analytics that underlie the pricing of bond and repo trades.

Revision Date	Comments
February 2022	First edition for version 17.0.

Contents

Section 1. Market Data Generation

1.1 Repo Curves

Simple Curve Definition – Repo curves are entered as basis point spreads over the discount curve.

Repo curves are associated with a pricer configuration in the Repo panel. Note that you can also choose to associate a discount curve for the Repo usage.

1.2 Discount Curves

Refer to the *Calypso Interest Rate Derivatives Analytics Guide* and *Yield Curves Generation Guide* for details.

Section 2. Bond and Repo Pricing

Product	Calypso Product	Description	Pricer
Fixed Coupon Bond	Bond	All major currencies supported (UST, JGB, OAT, GILT, CANADAS, BUND, etc.).	PricerBondBrazilian PricerBondBTAN PricerBondBTPS PricerBondBund PricerBondCCT PricerBondCGS PricerBondCMT PricerBondCanadas PricerBondGeneric PricerBondGilt PricerBondJGB PricerBondMarkkas PricerBondOAT PricerBondPOT PricerBondSTRIPS PricerBondUST
	BondAssetBacked		PricerBondAssetBacked
	BondBrady		PricerBondBrady
	BondConvertible		PricerBondConvertible
	BondHolding		PricerHolding
	BondMMDiscount		PricerBondMMDiscount
	BondMMDiscountAUD		PricerBondMMDiscountAUD
	BondMMInterest		PricerBondMMInterest
Issue	Issuance		PricerIssuance
Floating Rate Notes	BondFRN		PricerBondFRN
	Bond	UST floating rate note	PricerBondUSTFRN
Bond Option	BondOption		PricerBondOption

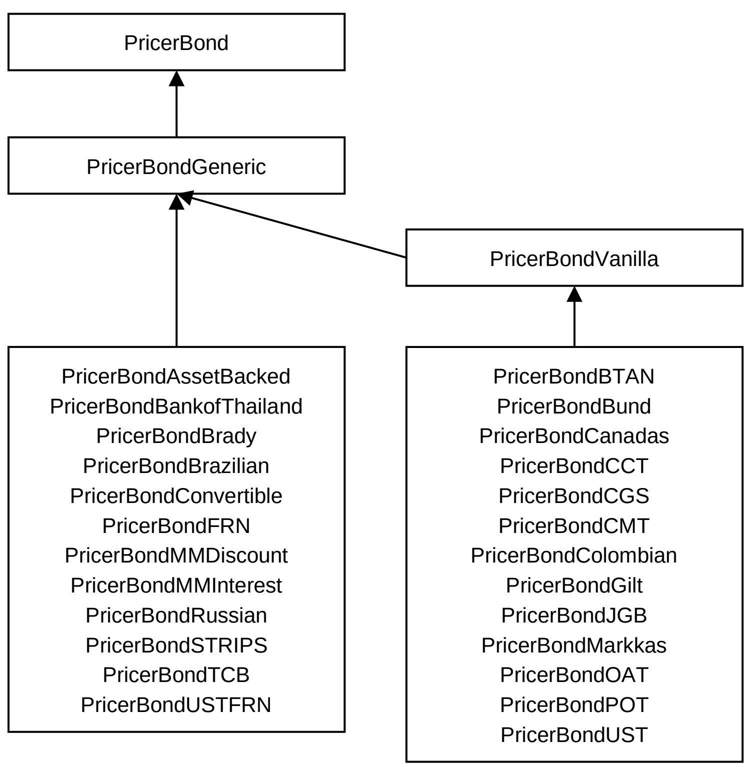
Product	Calypso Product	Description	Pricer
Repo	JGBRepo		PricerJGBRepo
	Repo		PricerRepo
Security Lending	SecLending		PricerSecurityLending
Treasury Lock	TreasuryLock	User locks in current bond yield until lockout date.	PricerTreasuryLock
Credit Linked Bond (*)	BondCLN	Standard	PricerBondCLN
		Basket	PricerBondCLNBasket

(*) For Credit Linked Bonds, please refer to the Calypso Credit Derivatives Analytics Guide for details.

2.1 PricerBond - PricerBondGeneric

This pricer covers general bond pricing functionality. Certain pricer measures are custom defined for particular currencies as outlined below.

Following is a simple diagram of the bond pricer structure. Arrows denote inheritance.



2.1.1 Pricer Measures

Dirty Price – The computation of Dirty Price depends on the parameter settings. If BOND_FROM_QUOTE=true, then the Dirty Price is just obtained from the price in the quote set, plus the accrued interest (the latter is calculated by the pricer using the Accrual Daycount of the bond). This would require a computation of price from yield if the quote type is YIELD (see Yield To Maturity), or a price from spread and a benchmark for quote type SPREAD. If BOND_FROM_QUOTE=false, the Dirty Price is equal to the Net Present Value (NPV) of the bond. The NPV is the sum of the present values of each of the cashflows, including coupon and principal:

$$Dirty\_\mathrm{Pr}ice = \sum_{i=n}^{N-1} C_i df_i + F df_N$$

- N = Total number of cashflows generated
- $0 \le n \le N-1$ n = Index of last coupon payment time
- C_i = Coupon payment for time index i
- F = Face value (redemption principal)
- df_i = Discount factor for time index i, which discounts a cashflow from time index i to the given settle date

The coupon payments C_i are computed as defined on the bond product, which can have been customized by the user. Coupon payments are computed using the bond's Coupon Daycount, in contrast to the Accrual Daycount used to compute accrued interest for pricing trades. To employ a standard bond with coupon payments that are all equal, the Coupon Daycount should be set to ACTB/ACTB on the bond. Using this method, a semiannual bond will have payments exactly equal to the coupon rate divided by 2, times the principal. For other coupon daycounts, the pricer uses that daycount convention to compute the amount of the payment for each period, which will vary according to the length of the period.

Clean Price – Clean Price = Dirty Price – Accrual

NPV – NPV = Dirty Price – Trade Dirty Price

calcNPV is called within the process function to calculate NPV. In a standard case, NPV is the sum of clean price and accrual. calcNPV uses method cleanPrice to calculate clean price and uses accrualInBondCurrency or getFinalCouponValueInBondCurrency to calculate accrual. The latter method is called if the maturity date falls between the valDate and the settle date; therefore the accrual should be the entire final coupon. The cleanPrice method can choose either implementing the algorithm presented in this document using repo curve / discount curve or using the market quote price. There is a variation in NPV calculation when ZeroDayPricing flag is set to true, which is not covered in this document.

NPV_NET – NPV_NET = Clean Price – Trade Clean Price

Yield to Maturity – Yield is the rate of discount at which the present value of the promised future cashflow equals the dirty price of the bond. When the Coupon Daycount is ACTB/ACTB, the standard yield-to-maturity is employed using equal-length discounting periods; for any other Coupon Daycount, actual time to each payment date is used in discounting (sometimes called "Actual Yield"). To calculate standard yield-to-maturity from a dirty bond price P_t, Calypso iteratively solves the following equation for y:

$$P_t = \sum_{i=n}^{N-1} \frac{C_i}{(1+\frac{y}{m})^{i+w}} + \frac{F}{(1+\frac{y}{m})^{N-1+w}}$$

If the evaluation date falls in the last cashflow period or the bond is a zero coupon bond, with CF the total last cashflow:

For USSTREET yield method:

$$P_t = \frac{CF}{1+\frac{y}{m}w}$$

For ISMA yield method:

$$P_t = \frac{CF}{(1+\frac{y}{m})^w}$$

In these formulas, *w* is the fractional period remaining from the settle date to the end of the current accrual period, using the Coupon Daycount (actual days if ACTB/ACTB).

Yield calculations are detailed below.

DURATION – Macaulay Duration, implemented based on the following relationship:

Macaulay Duration = (1 + y/n) * MODIFIED_DURATION, where y is the yield to maturity, n is the number of periods per year, and MODIFIED_DURATION is the percentage change to price with respect to change in yield.

We approximate MODIFIED_DURATION by using 1 basis point change to the yield.

MODIFIED_DURATION = - (dP_up – dP_down) / (2 * d) / price where d = 0.0001, and dP_up and dP_down are the price changes from modifying yield up and down by d, respectively.

CONVEXITY – The convexity of a bond is the change in the slope of the price-yield curve, for a small change in the yield. Following is the formula in PricerBondGeneric:

$$Convexity = \frac{P(y+h) - 2P(y) + P(y-h)}{P(y)h^2}$$

- P(y) is the price of bond as a function of yield y
- h represents a small movement of yield

2.1.2 Yield Calculations

Odd First and Last Coupons in Yield Calculations

The specifications in *Standards Securities Calculation Methods* (SIA, 1993) are applied to the handling of odd first and last coupons in the price-given-yield formulas for bonds.

In all yield formulas, discounting of each cashflow is done using a factor of the form

$$\frac{1}{(1+y/n)^{w+i}}$$

where *i* is the number of whole payment periods in the time interval between the valuation date to the cashflow payment date, and *w* is the remaining fractional period not covered by the whole periods. The odd coupon formulas serve to define *w*.

Discounting when Value Date Falls Before First Coupon

Starting at the first coupon payment date and working backwards, one first defines regular bond periods by subtracting the regular bond payment frequency repeatedly to generate hypothetical regular period end dates. These periods are termed "quasi-coupon periods".

The fractional period *w* is defined with respect to these quasi-coupon dates as follows:

$$w = q + \frac{d(t, Q_{end})}{d(Q_{start}, Q_{end})}$$

- t = value date
- Qstart = start of quasi-coupon period in which t falls
- Qend = end of quasi-coupon period in which t falls
- q = number of whole quasi-coupon periods falling between t and the first coupon payment
- d(x, y) = days between any given dates x and y, using the bond's day count convention

This formula covers both long and short first coupons. For a short first coupon, Q_{end} is always the coupon payment date, while for a long first coupon Q_{end} can be one of the earlier quasi-coupon dates. For the case where the first coupon is not odd, Q_{start} and Q_{end} are the normal start and end dates of the first coupon period, q=0, and the formula reduces to the usual case.

In the yield formula, this *w* is added to each *i* to define the discount factor for each cashflow, as given previously. For the first coupon *i* is 0, and so *w* itself gives the time for discounting the first payment. To discount an odd last coupon, an addition is made to *w*, as described in the following.

Discounting Last Coupon and Principal Payment

For either a normal, long, or short last coupon period, discount factor for the interest and the redemption payment at maturity is, for US Street yield

$$\frac{1}{1+\frac{y}{n}(w+N-1+v)}$$

or, if ISMA yield

$$\frac{1}{(1+y/n)^{w+N-1+v}}$$

The quantities in this formula are found as follows. First define T_M = Date of the maturity payment of the last coupon and the redemption principal.

Then in the formula we have

- N = number of coupon payment dates falling strictly after the value date t and strictly before T_M; if N = 0, set N equal to 1 in the formula
- w = the fractional period to the next coupon, if N > 1; 0 if N <= 1

Note that *w* is computed as in the preceding section if the value date falls before the first coupon date; else it is the ordinary fractional period remaining for a regular coupon.

Note also *w* is *not* included in the formula (it is zero) if the value date is in the last coupon period.

In the formula, the contribution from the last coupon period is

$$v = q + \frac{d(T_1, T_2)}{d(Q_{start}, Q_{end})} + X \frac{d(Q_{M,start}, T_M)}{d(Q_{M,start}, Q_{M,end})}$$

Quasi-coupon periods are computed starting from the bond coupon date prior to the maturity date T_M, and adding regular bond payment periods successively until reaching or passing T_M. Then

- *t* = value date
- q = number of whole quasi-coupon periods falling between the value date t and T_M
- T_1 = the later of *t* or the start date of the last coupon period
- Q_{start} = start of quasi-coupon period in which T_1 falls
- Q_{end} = end of quasi-coupon period in which T_1 falls
- T2 = Min(T_M, Q_{end})
- $Q_{M,start}$ = start of quasi-coupon period in which the maturity date falls
- $Q_{M,end}$= end of quasi-coupon period in which the maturity date falls
- X = 0 if $Q_{M,start} = Q_{start}$, and 1 otherwise

Note how the last term of *v* is only included if the maturity date and the value date do *not* fall in the same quasi-coupon period.

This formula covers normal, short, and long last coupons. If the last coupon period is of regular length, the above formula reduces to 1 if the value date falls before the start of the period, or reduces to the ordinary formula for fractional coupon periods otherwise.

Money Market Yield Methods

For MM_ACT360 or MM_ACT365 yield methods, different formulas are used.

If the time to maturity is less than or equal to one year:

$$P_t = \sum_{i=n}^{N-1} \frac{C_i}{(1 + y\frac{w+i}{m})} + \frac{F}{(1 + y\frac{w+N-1}{m})}$$

Otherwise:

$$P_t = \sum_{i=n}^{N-1} \frac{C_i}{(1+y)^{\frac{w+i}{m}}} + \frac{F}{(1+y)^{\frac{w+N-1}{m}}}$$

Calypso uses a solver to solve the nonlinear equation of price and yield. The solver is a slightly modified version of the Brent Method from pp. 268-269.

"Numerical Recipes in C" by Press, Flannery, Teukolsky, Vetterling. (In the 2nd edition this is p. 361, the Van Wijngaarden-Dekker-Brent Method.)

Money Market Yield Method for Monthly Yield Quoted Bonds

MM_ACT30 Yield Method

This yield method is currently only supported for zero coupon bonds.

Several money market securities use a monthly yield quote instead of an annual yield quote. These monthly quotes use an ACT/30 daycount.

$$PV = \frac{1}{1 + Yield_M \cdot dcf} * FV$$

- *PV* = present value
- *FV* = face value at maturity
- $Yield_M$ = monthly yield quoted
- *dcf* = d / 30 = daycount fraction using 30 days as base (ACT/30)
- *d* = number of calendar days between the date of valuation and the date of maturity

Exponential Yield Method

Yield Method = Exp_ACT360 or Exp_ACT365

These yield methods use the ACT/360 or ACT/365 daycount conventions for computing the year fraction, irrespective of the bond's daycount.

The following formulas are used for discounting:

Exp_ACT360:

$$\frac{1}{(1 + IRR)^{\frac{ACT}{360}}}$$

Exp_ACT365:

$$\frac{1}{(1 + IRR)^{\frac{ACT}{365}}}$$

- *IRR* = annual rate
- *ACT* = actual number of days in the year

Some uses are Peruvian inflation linked bonds (Exp_ACT360), and Chilean bonds (both).

Yield Method = Exponential

Same as above with the bond daycount instead of a hard-coded daycount.

$$\frac{1}{(1 + IRR)^{Bond\ Daycount}}$$

Yield Method = Exp_NL365

Same as Exp_ACT365 except leap year is not considered.

Thai Bond Market Yield Method

Yield Method = ThaiBMA

PricerBondGeneric is used to price fixed rate Thai bonds, as well as “Non-Bank of Thailand FRN” Thai bonds, bond type = Generic.

For “Bank of Thailand FRN” Thai bonds, see PricerBondBankofThailand.

Fixed Rate Thai Bonds

In the event of the last period being irregular and settlement not falling into the last period, the following formula is used:

$$Dirty\ Price = \sum_{t=0}^{n-2} \frac{CF_t}{\left(1 + \frac{\gamma}{100 * h}\right)^{\left(t + \frac{DSC * h}{365}\right)}} + \frac{CF_{last}}{\left(1 + \frac{\gamma}{100 * h}\right)^{\left((n-2) + \frac{(DSC + DCD) * h}{365}\right)}}$$

DSC: Number of actual days from settlement date to next coupon date

DCD: Number of actual days in last coupon period

Otherwise the following formula should be used:

$$Dirty\ Price = \sum_{t=0}^{n-1} \frac{CF_t}{\left(1 + \frac{\gamma}{100 * h}\right)^{\left(t + \frac{DSC*h}{365}\right)}}$$

"Non-Bank of Thailand FRN" Thai Bonds

$$price * \left[\left(1 + \frac{(i_{trade} + dm)}{h}\right)^{f1}\right] = k + \sum_{t=1}^{n-1} \frac{i_{trade} + qm}{h} * V^t + 100 * V^{n-1}$$

$$V = \frac{1}{1 + \frac{(i_{trade} + dm)}{h}}$$

i_{trade}: fixed to the index rate on the trade date. Note: If the time is before the publish time specified in the index rate definition (Time Zone, Hour fields), then (date – 1) is used instead.

Note that no other terms in the formula should have their values changed. For example, *k* should remain the same value, which is based on the value of the index rate on the last reset. In addition, the accrued interest calculation should not be changed.

In the event that the trade date is on the issue date of the bond, the pricing formula changes to:

$$price * \left[\left(1 + \frac{(i_{trade-2} + dm)}{h}\right)^{f1}\right] = k + \sum_{t=1}^{n-1} \frac{i_{trade-2} + qm}{h} * V^t + 100 * V^{n-1}$$

$$V = \frac{1}{1 + \frac{(i_{trade-2} + dm)}{h}}$$

$i_{trade-2}$: fixed to the index rate on (trade date – 2). Note: If the time is before the publish time specified in the index rate definition (Time Zone, Hour fields), then (date – 1) is used instead.

In addition, the accrued interest and *k* should be based on $i_{trade-2}$.

Daycount

Bonds with the ThaiBMA yield method use the hard-coded daycount 30Thai/360 to price.

30Thai/360 does special handling for the last day of February. For all other months, it is the same as daycount 30E/360.

– If D2 falls on the last day of February and D1 falls on the 29th, then change D2 to the 29th
– If D1 falls on the last day of February and D2 falls on the 29th, then change D1 to the 29th

If the bond has an irregular period, it uses the daycount defined in the bond setup for that period. If the daycount is customized in the cashflows, it uses the custom daycount for the irregular period.

In the case that the bond is amortizing, it is priced using the daycount ACT/365. Note that this is for pricing only, and not payment generation.

This does not apply to the accrual daycount. The accrual daycount is the acc daycount defined in the bond setup.

MODIFIED_DURATION

Thai FRNs using the yield method ThaiBMA and either PricerBondGeneric or PricerBondBankofThailand approximate MODIFIED_DURATION by bumping both the yield and the future cashflows, rather than just the yield.

Mexican Yield Method

Yield Method = MXN

Discounting Daycount

For the MXN yield method, if the frequency is QTR or SA, then the discounting daycount is 91 or 182, respectively. This does not apply to the coupon daycount. However, if "Use Date Rule" is selected on the Coupon panel, and the date rule specified is of type ADD_PERIOD, then the number of days will be taken from the date rule instead of from the frequency.

Fixed Rate Mexican Bonds (MBONO, UDI)

Yield Method = MXN

Coupon Type = Fixed

The formula to calculate the dirty price is as below:

$$price = \left(\frac{\left[c + c * \left[\frac{1}{R} - \frac{1}{R * (1+R)^{K-1}} \right] + \frac{VN}{(1+R)^{K-1}} \right]}{\left[(1+R)^{1-\frac{d}{m}} \right]} \right) - c * \frac{d}{m}$$

- VN = face value of the security in MXN
- d = number of days between the issue date or the last interest payment date, whichever is applicable and the valuation date
- m = number of days in a coupon period
- c = coupon payment = VN * [(m*annual coupon)/360]
- R = YTM * (m/360)
- K = number of coupon payments remaining after valuation date

Floating Rate Mexican Bonds (BPA)

Yield Method = MXN

Coupon Type = Floating

Coupon Frequency ≠ DLY

BPA bonds are Mexican floating rate bonds with the following calculations:

Future Coupons are set to the reference rate based on the Valuation Date, not the Reset Date.

$$Discount\ Factor = \frac{1}{\left(1 + Yield * \frac{Days_{Cpn}}{360}\right)^{\frac{Days_{Val\ to\ Coupon\ End}}{Days_{Cpn}}}}$$

– $Days_{Cpn}$ = number of days in the coupon

– $Days_{Valuation\ to\ Coupon\ End}$ = number of days from valuation date to coupon end date

Most Mexican floating bonds have 28D, 91D, or 182D frequencies, so the number of days would be the same for all coupons.

Floating Rate Mexican Bonds with Daily Compounding (BONDE, BREM)

Yield Method = MXN

Coupon Type = Floating

Coupon Frequency = DLY

BONDESD and BREMS bonds are Mexican floating rate bonds with the following calculations:

$$Discount\ Factor = \left[\frac{1}{\left(1+\frac{Yield}{360}\right)^{Days_{Cpn}}}\right]^{\frac{Days_{Valuation\ to\ Coupon\ End}}{Days_{Cpn}}}$$

– $Days_{Cpn}$ = number of days in the coupon

– $Days_{Valuation\ to\ Coupon\ End}$ = number of days from valuation date to coupon end date

BONDESD and BREMS have 28D, 91D, or 182D frequencies, so the number of days would be the same for all coupons.

Note: BONDEST and BONDES182 do not use these calculations; instead, their calculations are similar to BPA.

Zero Coupon Mexican Bonds (CETE, PAGARE, Interest and Principal Strips)

Bond Class = BondMMDiscount

Yield Method = MXN

Some zero coupon Mexican bonds such as Strips require the following calculation:

$$Discount\ Factor = 1 + Yield * \frac{Days}{360}$$

– Days = number of days from valuation date to coupon end date

Note: This yield method is similar to MM_ACT360, except that it uses simple compounding regardless of the time to maturity.

Norwegian Yield Method

This yield method is used for Norwegian bonds. It uses a yield discounting daycount similar to ACTB/ACTB, except that the first fractional period is calculated on an ACT/365 basis rather than ACT/ACT.

Argentine Yield Method

Yield Method = ARS

This yield method is used for Argentine dual-currency bonds. It uses the following discount formula:

$$Discount\ Factor = \frac{1}{\left(1 + Yield * \frac{PeriodDays}{YearDays}\right)^{\frac{Days}{PeriodDays}}}$$

– Days = actual (ACT) number of days
– YearDays = uses the base of the coupon daycount
– PeriodDays = uses the following values:

Coupon Frequency	PeriodDays
MTH	30
QTR	90
SA	180

2.1.3 Risky Bond Pricing

For a risky bond, Calypso applies both a credit curve and interest rate curve to price it. When the issuer defaults, no accrual coupon payment is received, only the principal recovery amount is received.

i) **If pricing parameter ZD_PRICING is true, the formula to price a risky bond is**

$$NPV = \sum_{i=1}^{n} Coupon_i * DF_i * q_i + N * DF_i * (q_{i-1} - q_i) * R + N * DF_n * q_n$$

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, and $q_0 = 1$,

N is the nominal value of the risky bond,

n is the number of remaining coupon payments,

R is the recovery rate of the risky bond.

ii) **If pricing parameter ZD_PRICING is false, the valuation formula of the risky bond becomes**

$$NPV = \frac{1}{q_{se} * DF_{se}} \sum_{i=1}^{n} Coupon_i * DF_i * q_i + N * DF_i * (q_{i-1} - q_i) * R + N * DF_n * q_n$$

Where

DF_{se} is the discount factor on the settlement date,

q_{se} is the survival probability on settlement date.

2.1.4 Pricing Parameters

Calypso uses Pricer Parameters to customize trades. Pricer Parameters can either be set in the pricing environment or be set in the trade window. When set in the trade window, PricerParameters are "transient or local", which means they would not be known by other similar trades or even the clients of the trades. For example, risk analyses which are invoked outside the trade window would use the "static or global" parameter setting in pricing environment.

Following are the descriptions of some parameters related to bond pricers.

- BOND_FROM_QUOTE
  - Values — True or False.
  - Usage — When set to true, the bond will be priced from quote, such as clean price, quote, discount, etc.
- REPO_RATE
  - Values — Double.
  - Usage — If the valuation date is before the settle date of the bond, Calypso is going to get a repo rate for forward pricing. The repo rate can come from the pricing parameters. If it is not set there, then Calypso will need to find the discount and repo curves in order to produce the repo rate.
- INCLUDE_FEES
  - Values — True or False.
  - Usage — If set to true, the pricer measures such as NPV and FEES_NPV will reflect the actual fees involved in the trade.
- ZD_PRICING
  - Values — True or False.
  - Usage — If set to true, the bond price will be discounted to the valuation date from settle date.
- CHECK_FUNDING_RATES
  - Values — True or False.
  - Usage — If set to true, the pricer will check if funding rates are set or implemented.
- NPV_INCLUDE_CASH
  - Values — True or False.
  - Usage — If set to true, and if there is cashflow on valuation date, the pricer will add/subtract the cash.
- NPV_INCLUDE_COST
  - Values — True or False.
  - Usage — If set to true, the NPV will reflect settlement cost.

2.2 PricerBondBankofThailand

PricerBondBankofThailand is used to price "Bank of Thailand FRN" Thai bonds.

Yield Method = ThaiBMA

Bond Type = BankofThailand

$$price * \left[1 + \left(i_{settle\_to\_coupon} + dm\right) * \frac{f1}{h}\right] = k + \sum_{t=1}^{n-1} \frac{i_{trade} + qm}{h} * V^t + 100 * V^{n-1}$$

$$V = \frac{1}{1 + \frac{(i_{trade} + dm)}{h}}$$

i_{trade}: Fixed to the index rate on the trade date. Note: If the time is before the publish time specified in the index rate definition (Time Zone, Hour fields), then (date – 1) is used instead.

In the case of being in the XI Period,

$$i_{settle\_to\_coupon} = i_{trade}$$

Otherwise

$$i_{settle\_to\_coupon} = i_{interpolated}$$

Interpolation Interest

In the event that $i_{interpolated}$ in needs to be calculated:

$$i_{interpolated} = i_{shorter} + \left(i_{longer} - i_{shorter}\right) * \frac{days_{tocoupon}}{days_{tenordiff}}$$

$i_{shorter}$: Interest for the index using the nearest tenor shorter than (trade date + 2) to the next coupon date

i_{longer}: Interest for the index using the nearest tenor longer than (trade date + 2) to the next coupon date

$days_{tocoupon}$: Number of days from (trade date + 2 + shorter tenor) to the next coupon date

$days_{tenordiff}$: Number of days from (trade date + 2 + shorter tenor) to (trade date + 2 + longer tenor)

Note: When adding tenors to a date, they are done in weekly, monthly, etc. increments. For example, if the valuation date is 8/14/2013 and the tenor is 3M, then (valuation date + 2 + tenor) = (8/16/2013 + tenor) = 11/16/2013.

The interest rates are valued as of (trade date + 2).

FRN Rate Fixing - Auction Period

In the event that the trade date is on or before the issue date of the bond, or the trade attribute ReOpen is set to true, the pricing formula changes to:

$$price * \left[1 + (i_{trade-2} + dm) * \frac{f1}{h}\right] = k + \sum_{t=1}^{n-1} \frac{i_{trade-2} + qm}{h} * V^t + 100 * V^{n-1}$$

$$V = \frac{1}{1 + \frac{(i_{trade-2} + dm)}{h}}$$

$i_{trade-2}$: Fixed to the index rate on (trade date – 2). Note: If the time is before the publish time specified in the index rate definition (Time Zone, Hour fields), then (date – 1) is used instead.

In addition, the accrued interest and k should be based on $i_{trade-2}$.

2.3 PricerBondBrady

Brady bonds are obligations of the governments of developing countries with partial guarantees. Therefore they bear more default risk than US treasuries. This is reflected in PricerBondBrady class' yield function, which is the only function implemented for this class. The only difference between this function and the one in its grandparent PricerBond is that it checks whether or not the bond defaults before evaluation date and returns zero if it did default.

2.4 PricerBondBund

Pricer for German government bonds.

The dirty price method uses the Yield Method ISMA or MOSS according to the setup of the bond. If the Yield Method is set to ISMA, we use the formulae defined in PricerBondGeneric.

2.5 PricerBondCanadas

Pricer for Canadian government bonds.

Yield method = BankOfCanada

When only one coupon remains: P = (100 + C/2)/[1 + Y*T/(day type)] – A

Otherwise

$$P = \left(\sum_{i=n}^{N-1} \frac{CF_i}{(1+\frac{Y}{m})^i}\right) \times \left(1 + Y \times \frac{x-z}{Year}\right)$$

- P_t is the bond quote/dirty/invoice price at settlement time t
- N is the total number of cashflows generated by bond
- $0 \le n \le N-1$ n is the index of last coupon payment time relative to t
- C_i is the coupon for time index i
- F is the notional principal
- m is the coupon frequency
- z is the number of days between the settlement date and the next cashflow date
- x is the number of days between the two cashflows

2.6 PricerBondCGS

Pricer for Australian government bonds.

Calculates the dirty price using the RBA bond formula:

$$P = v^{f/d}(g(c+a) + v^n)$$

- v = 1/(1+i) where i is the annual yield / coupons per year
- f = days from settlement to the next interest payment date
- d = days in the current coupon period
- c = 1 if the bond is cum interest, 0 otherwise
- g = annual coupon rate / coupons per year
- n = the number of full coupon periods between the next interest payment date and the date of maturity (i.e. number of coupons - 1)
- $a = (1-v)^n / i$

Bonds with only their final coupon payment remaining are priced using a simple discount formula:

$$Price = \frac{1+g}{1+\left(\frac{f}{365}*i\right)}$$

On the last coupon date, the rounding of clean/dirty can be controlled by configuring the pricing parameter CGS_LAST_CPN_ROUNDING. It is the number of decimal places. It defaults to 10.

2.7 PricerBondColombian

Pricer for Colombian non-government floating rate securities linked to the DTF, IBR, or IPC indices.

2.7.1 DTF-linked Bonds

To calculate the coupon, the DTF rates are converted to nominal rates, NATA. This is required to add the spread. Once the spread is added, the rates are converted back to effective annual (EA) rates. The effective annual rate is then used for computing the cashflows.

The rate conversions are defined by the following formulas:

$$NATA\ Rate = ROUNDING\left(\left(\frac{(1+DTF_{EA})^{1/4}-1}{1+\left((1+DTF_{EA})^{1/4}-1\right)}\right)*4\,,4\right)$$

$$EA\ Rate = \left(1-\frac{NATA\ Rate + Spread}{4}\right)^{-4}-1$$

The TV Rate (quarterly in arrears) is then used for calculating the projected cashflows:

$$\text{TV Rate} = (1+\ \text{EA Rate})^{\text{No.of days between flows}/\text{Base}}-1$$

2.7.2 IBR-linked Bonds

For IBR-linked bonds, the rate and spread are both in nominal terms, so no rate conversion is required.

The nominal rate is used for cashflow calculations:

$$\text{Coupon Rate}_{\text{Term}} = \frac{\text{IBR}_{\text{Valuation Date}} + \text{Spread}}{12}$$

For calculating projected cashflows:

$$Cashflow = \text{Coupon Rate}_{\text{Term}} * \text{Nominal amount}$$

$$\text{Cashflow}_{\text{Final}} = \text{Coupon Rate}_{\text{Term}} * \text{Nominal Amount} + \text{Nominal Amount}$$

For NPV calculation:

$$PRICE = \text{PV}\left(\text{IBR}_{\text{Valuation DAte}}, \frac{\text{DV}}{\text{Basis}},, -(\text{Coupon Rate}_{\text{Term}} + \text{Nominal }\%)\right)$$

$$NPV = \text{PRICE} * \text{NominaL Amount}$$

2.7.3 IPC-linked Bonds

$$\textit{Periodic coupon} = [[((1+\text{ EA inflation rate}) * (1+\text{Spread}))\text{^}(\text{base days}/365)] - 1]$$

2.8 PricerBondFRN

Floating Rate Note is a fixed principal instrument, often utilized as the basis for a swap which has a long or even indefinite life and whose yield is reset periodically relative to a reference index rate to reflect changes in short- or intermediate-term interest rates.

PricerBondFRN extends PricerBondGeneric and the only difference is a custom algorithm in PricerBondFRN to solve for the dirty price from yield to maturity. We assume that the bond matures on the next coupon date when computing the dirty price from the yield.

2.9 PricerBondGilt

PricerBondGilt implements the simplified formulae published by the UK Debt Management Office to compute the dirty price and yield of perpetual Gilt bonds.

2.10 PricerBondJGB

Pricer for Japanese government bonds.

For this pricer, there are BOJ conventions that we want to follow when the user indicates that the conventions are desired by setting the BOJ flag in the pricing environment for the duration of pricing. After pricing, unset the flag if it wasn't set before.

A numeric duration method overrides the analytic formula in PricerBond. The denominator in the formula is the clean price instead of the dirty price.

The accrual method for a BondJGB is redefined due to the fact that these bonds have a different daycount convention for Coupon and Accrual. When defining a BondJGB, the user should set up the daycount to ACTB/ACTB. In this function, the Accrual is computed by the following formula: Coupon Rate * Number of days between Start Date and accrual using ACT_365 basis.

The Yield of a BondJGB is computed based on the Simple Yield Formula.

2.11 PricerBondMMInterest

PricerBondMMInterest implements the simple money market yield formula for coupon bearing money market notes represented by the BondMMInterest product in Calypso.

$$P = \left(\sum_{i=1}^{n} \frac{c/h}{DF_i} + \frac{100}{DF_n} \right).$$

$$DF_i = DF_{i-1} \cdot (1 + y_m \cdot d/a) \text{ for } i = 2 \text{ to } n$$

$$DF_1 = 1$$

P = dirty price (clean price plus accrued interest) of the bond per 100 units face value

y_m = annual money market yield (simple interest)

c = annual percentage coupon rate

n = number of coupon periods to maturity

h = number of coupon periods in a year

d = number of days in the coupon period using the bond coupon day count convention

a = number of days in a year using the bond coupon day count convention

2.12 PricerBondRussian

Pricer for Russian inflation bonds. It uses the following accrual calculations.

$$AI_{im} = N_i * CPN * (i - t_m) / 365 \text{ (it is rounded to 2 decimals)}$$

where

t_m = start date of current coupon period

$$N_i = N_{base} * I_i \text{ (it is rounded to 2 decimals)}$$

where

N_{base} = issue nominal of bond

$I_i = INDEX_i / INDEX_{base}$

$$INDEX_I = Ref\_CPI_{M(i)-4} + (Ref\_CPI_{M(i)-3} - Ref\_CPI_{M(i)-4}) * (n - 1) / d(i)$$

where

i = calendar date

$INDEX_{base}$ = inflation index on Issue Date

n = order number of i (calendar date) in the settle month

$d(i)$ = number of days in settle month

2.13 PricerBondUSTFRN

2.14 PricerIssuance

Issuance is used to model the issuance of a bond or any security papers by a processing organization. Keep in mind that the pricer measures are evaluated from the issuer's perspective, not the investor.

PricerIssuance extends Pricer class. However, unlike other pricer classes such as PricerBond, its price method does not call a process method to calculate the measures. Instead it delegates the task to the product/trade's price method. PricerIssuance itself doesn't have a process method. Therefore you would not see some market data information such as discount curve in the pricing window as in other pricer windows since everything happens in the background.

PricerIssuance may produce a different value from that produced by the product/trade's pricer class for the same pricer measure. The difference comes from the fact that the cashflow patterns for issuance organization and investor are opposite or at least different. For example, the bond issuer has positive cashflow at first, then negative cashflows in rest of the bond lifecycle, while the bond investor has negative cashflow at first, then positive cashflows in rest of the time. That explains that for the same bond, issuance pricer and bond pricer agree on dirty price and accrual but not on duration, convexity and NPV, etc.

2.15 PricerRepo

PricerRepo delegates the pricing to the pricer of the underlying instrument. The NPV is the value of collateral minus the cost of funding (cash leg of repo).

PricerRepo requires a discount curve and if the trade is floating, it needs a forecast curve.

The following pricers extend PricerRepo for the corresponding types of repos: PricerJGBRepo and PricerPledge.

2.15.1 Pricing Parameters

The following pricing parameters apply to repo pricing:

- BOND_FROM_QUOTE
  - Values — True or False.
  - Usage — Should be True for bond and repo. It means that pricing of underlying product (bond) will be done from the closing price instead of using a curve.
- NPV_INCLUDE_COST
  - Values — True or False.
  - Usage — Should be set to True for bond pricing where settlement amount is included in the bond NPV pricer measure. If it is True for repo as well, the cash side of the repo is included in the repo NPV.
- NPV_INCLUDE_COST_AFTER_SETTLE_DATE
  - Values — True or False.
  - Usage — Should be set to False for bond pricing. It means that once the bond trade is settled, the settlement amount is no longer included in the NPV.
- ZD_PRICING
  - Values — True or False.
  - Usage — If True for bond, the price is discounted until today instead of settlement date. If True for repo, the cash side of the repo is discounted the maturity date and today instead of settlement date.

2.15.2 Repo NPV

Value of collateral minus the cost of funding (cash leg of repo). The following cases are supported:

- If INCLUDE_NPV_COLLAT=False

  NPV = Sum of all discounted future flows + cost

  where cost:

  - If NPV_INCLUDE_COST=False:

    cost = 0
  - If NPV_INCLUDE_COST=True:

    cost = sum of all PRINCIPAL flows in the past
- If INCLUDE_NPV_COLLAT=True

  NPV = (Bond trade DIRTY_PRICE at repo end date x nominal) – cost

  where cost:

  - If NPV_INCLUDE_COST=False

    cost = 0
  - If NPV_INCLUDE_COST=True

```
cost = (sum of all repo flows excluding initial principal)
```

The calculation of UNDERLYING_FWD_PRICE depends on BOND_FROM_QUOTE and the bond's DIRTY_PRICE computed by the bond pricer. The bond's DIRTY_PRICE is calculated using collateral clean price and accrual traded with a settlement date equal to end date of repo trade.

For the cash side of the repo, the system sums all pure cashflows (not collateral flows) where the settlement date is not equal to the repo start date.

Those cashflows are discounted as of Today if ZD_PRICING for repo is set to True.

Also, the funding side is used in the calculation of repo NPV only if NPV_INCLUDE_COST for repo is set to true.

2.15.3Repo PV_COLLAT

PV of the trade's collaterals. Same as PV pricer measure for the same bond/equity trade as the collateral with settle date = repo start date.

2.15.4Repo NPV_COLLAT

NPV of the trade's collaterals.

2.15.5Repo DIRTY_PRICE

Sum of PRICE and ACCRUAL_BO coming from bond pricer. In other words, it corresponds to the clean price on valuation date plus accrual gained from start date of the repo.

2.15.6Repo NPV_NET

This pricer measure is not implemented on the PricerRepo; it returns the generic NPV_NET pricer measure: NPV – ACCRUAL_BO where ACCRUAL_BO is interest accrued on the cash side of the repo.

Section 3. Bond Options Pricing

3.1 European Bond Options

PricerBondOption is based on Black's model for European style bond options.

The assumption made here is that the forward bond price has a constant volatility.

$$C = e^{-r_p t_p}(FN(d_1) - KN(d_2))$$

$$P = e^{-r_p t_p}(KN(-d_2) - FN(-d_1))'$$

where

$$F = \frac{Se^{-q^{t_e}}}{e^{-r_g t_e}} = Se^{(r_g - q)t_e}$$

and

$$d_1 = \frac{ln\left[\frac{F}{K}\right]}{\sigma\sqrt{t_e}} + \frac{\sigma\sqrt{t_e}}{2}$$

and

$$d_2 = d_1 - \sigma\sqrt{t_e}$$

This particular variation on the Black-Sholes model accounts for the timing of cashflows encountered in real work transactions. The model does this by using different time periods for each type of cashflow. The variables are described as follows:

- S = the spot price of the underlying security
- K = the strike price
- r_p = the continuously compounded risk-free rate with base ACT/365
- q = the continuously compounded dividend yield with base ACT/365
- r_g = the continuously compounded growth rate with base ACT/365

  Typically, periodically compounded rates which can be changed to continuous rates using the following equation:

  $$e^r = \left(1 + \frac{r}{n}\right)^{\frac{1}{n}}$$

  with n being the periodicity
- σ = the volatility of returns of the underlying security
- t_e = the time period from the valuation date to the option's expiration date, i.e. the time for which the option is traded
- t_p = the time to payment, i.e. from the valuation date to the settlement date (usually two days after expiration)
- N(x) = the cumulative standard normal distributive function
- r_p = the risk free rate
- r_g = 0

- q = 0

3.1.1 Call Options

NPV

NPV Bare = $\boldsymbol{e^{-rT_p}\{ FN(d_1) - KN(d_2)\}}$

NPV = NPV Bare * Notional - Premium

$$C = e^{-r_p t_p}(FN(d_1) - KN(d_2))$$

where

$$F = Se^{(r_g - q)t_e}$$

$$d_1 = \frac{\ln\left[\frac{F}{K}\right]}{\sigma\sqrt{t_e}} + \frac{\sigma\sqrt{t_e}}{2}$$

$$d_2 = d_1 - \sigma\sqrt{t_e} \qquad (1)$$

$$d_2^2 = d_1^2 - 2\sigma d_1\sqrt{t_e} + \sigma^2 t_e$$

$$d_2^2 = d_1^2 - 2\sigma\frac{\ln\left[\frac{F}{K}\right] + \sigma^2 t_e}{\sigma\sqrt{t_e}}$$

$$d_2^2 = d_1^2 - 2\ln(\frac{F}{K}) \qquad (2)$$

$$
\begin{aligned}
n(d_2) &= \frac{e^{\frac{-d_2^2}{2}}}{\sqrt{2\pi}} \\
&= \frac{e^{\frac{-d_1^2}{2}+\ln(\frac{F}{K})}}{\sqrt{2\pi}} \\
&= \frac{Fe^{\frac{-d_1^2}{2}}}{K\sqrt{2\pi}} \\
&= n(d_1)\frac{F}{K} \qquad\qquad (3)
\end{aligned}
$$

Delta

Delta Bare = $\boldsymbol{e^{-rT_p}\,N(d_1)}$

Delta = Delta Bare * Notional * F * 0.01

Derivation

$$\Delta_c = \frac{\partial C}{\partial S} = e^{-r_p t_p}\left(e^{(r_g-q)t_e}\left(Sn(d_1)\frac{\partial d_1}{\partial S} + N(d_1)\right) - Kn(d_2)\frac{\partial d_2}{\partial S}\right)$$

Substituting equation (3) gives :

$$\Delta_c = e^{-r_p t_p+(r_g-q)t_e}N(d_1) + e^{-r_p t_p+(r_g-q)t_e}Sn(d_1)\frac{\partial d_1}{\partial S} - e^{-r_p t_p}Kn(d_1)\frac{\partial d_1}{\partial S}\frac{F}{K}$$

$$\Delta_c = e^{-r_p t_p+(r_g-q)t_e}N(d_1) + e^{-r_p t_p+(r_g-q)t_e}Sn(d_1)\frac{\partial d_1}{\partial S} - Se^{-r_p t_p+(r_g-q)t_e}n(d_1)\frac{\partial d_1}{\partial S}$$

$$\frac{\partial d_1}{\partial S} = \frac{\partial}{\partial S}\frac{(\ln Se^{-qt_e} - \ln Ke^{-rt_e})}{\sigma\sqrt{t_e}} = \frac{1}{S\sigma\sqrt{t_e}}$$

$$\Delta_c = e^{-r_p t_p+(r_g-q)t_e}N(d_1)$$

Gamma

Gamma Bare = $\boldsymbol{e^{-rT_p} * \frac{n(d_1)}{F*\sigma\sqrt{T_e}}}$

Gamma = Gamma Bare * Notional * $\boldsymbol{(F*.01)^2}$

Derivation

$$\Gamma_c = \frac{\partial^2 C}{\partial S^2} = \frac{\partial}{\partial S}\Delta_c$$

$$\Gamma_c = e^{-r_p t_p + (r_g - q)t_e} n(d_1)\frac{\partial d_1}{\partial S}$$

$$\frac{\partial d_1}{\partial S} = \frac{1}{S\sigma\sqrt{t_e}}$$

Combining the above equations gives :

$$\Gamma_c = \frac{1}{S\sigma\sqrt{t_e}}\left(e^{-r_p t_p + (r_g - q)t_e} n(d_1)\right)$$

Vega

Vega Bare = $\boldsymbol{e^{-rT_p} * F * n(d_1) * \sqrt{T_e}}$

Vega = Vega Bare * Notional * 0.01

Derivation

$$\nu_c = \frac{\partial C}{\partial \sigma} = e^{-r_p t_p + (r_g - q)t_e} Sn(d_1)\frac{\partial d_1}{\partial \sigma} - e^{-r_p t_p} Kn(d_2)\frac{\partial d_2}{\partial \sigma}$$

$$\nu_c = e^{-r_p t_p + (r_g - q)t_e} Sn(d_1)\frac{\partial d_1}{\partial \sigma} - Ke^{-r_p t_p} n(d_1)\frac{\partial d_2}{\partial \sigma}\frac{S}{K}e^{r_g t_e - q t_e}$$

$$\nu_c = Se^{-r_p t_p + (r_g - q)t_e} n(d_1)(\frac{\partial d_1}{\partial \sigma} - \frac{\partial d_2}{\partial \sigma})$$

where

$$\frac{\partial d_1}{\partial \sigma} = -\frac{\ln\left[\frac{F}{K}\right]}{\sigma^2\sqrt{t_e}} + \frac{\sqrt{t_e}}{2}$$

$$\frac{\partial d_2}{\partial \sigma} = -\frac{\ln\left[\frac{F}{K}\right]}{\sigma^2\sqrt{t_e}} - \frac{\sqrt{t_e}}{2}$$

$$\nu_c = e^{-r_p t_p + (r_g - q)t_e} S\sqrt{t_e}\, n(d_1)$$

Rho

Rho Bare = $\boldsymbol{T_p * e^{-rT_p}\{K * N(d_2) - F * N(d_1)\}}$

$\mathbf{rho_{r_p}}$

$$\rho_{c,p} = \frac{\partial C}{\partial r_p} = e^{-r_p t_p} K t_p N(d_2) - t_p e^{-r_p t_p + (r_g - q)t_e} SN(d_1)$$

Rho3 Bare = $\boldsymbol{T_e * e^{-rT_p} * F * N(d_1)}$

$\mathbf{rho_{r_g}}$

$$\rho_{c,g} = \frac{\partial C}{\partial r_g} = e^{-r_p t_p + (r_g - q)t_e} Sn(d_1)\frac{\partial d_1}{\partial r_g} + St_e e^{-r_p t_p + (r_g - q)t_e} N(d_1) - Ke^{-r_p t_p} n(d_2)\frac{\partial d_2}{\partial r_g}$$

$$d_1 = \frac{\ln Se^{-qt_e} - \ln Ke^{-r_g t_e}}{\sigma\sqrt{t_e}} + \frac{\sigma\sqrt{t_e}}{2}$$

$$\frac{\partial d_1}{\partial r_g} = \frac{\partial d_2}{\partial r_g} = \frac{t_e}{\sigma\sqrt{t_e}}$$

Substituting equation (3) gives :

$$\rho_{c,g} = \frac{t_e e^{-r_p t_p + (r_g - q)t_e} Sn(d_1)}{\sigma\sqrt{t_e}} + St_e e^{-r_p t_p + (r_g - q)t_e} N(d_1) - Ke^{-r_p t_p} n(d_2)\frac{\partial d_2}{\partial r}$$

$$\rho_{c,g} = e^{-r_p t_p + (r_g - q)t_e} S(\frac{t_e n(d_1)}{\sigma\sqrt{t_1}} + t_e N(d_1)) - Ke^{-r_p t_p} n(d_2)\frac{\partial d_2}{\partial r}$$

$$\rho_{c,g} = e^{-r_p t_p + (r_g - q)t_e} S(\frac{t_e n(d_1)}{\sigma\sqrt{t_1}} + t_e N(d_1)) - e^{-r_p t_p} Kn(d_1)\frac{\partial d_1}{\partial r}\frac{S}{K}e^{(r_g - q)t_e})$$

$$\rho_{c,g} = e^{-r_p t_p + (r_g - q)t_e} S(\frac{t_e n(d_1)}{\sigma\sqrt{t_e}} + t_e N(d_1)) - S e^{-r_p t_p + (r_g - q)t_e} n(d_1) \frac{\partial d_1}{\partial r}$$

$$\rho_{c,g} = e^{-r_p t_p + (r_g - q)t_e} S t_e N(d_1)$$

Rho = (Rho Bare + Rho3 Bare) * Notional * 0.01

Theta

Theta Bare = $-\boldsymbol{\sigma} * \boldsymbol{F} * \boldsymbol{e}^{-\boldsymbol{r}\boldsymbol{T_p}} * \frac{\boldsymbol{n}(\boldsymbol{d_1})}{2*\sqrt{\boldsymbol{T_e}}} + \boldsymbol{r} * \boldsymbol{e}^{-\boldsymbol{r}\boldsymbol{T_p}} * [\boldsymbol{F}\boldsymbol{N}(\boldsymbol{d_1}) - \boldsymbol{K}\boldsymbol{N}(\boldsymbol{d_2})]$

Theta = Theta Bare * Notional / nDays

where nDays = number of days in a year

3.1.2 Put Options

NPV

NPV Bare = $e^{-rT_p}\{ KN(-d_2) - FN(-d_1)\}$

NPV = NPV Bare * Notional - Premium

Delta

Delta Bare = $e^{-rT_p} N(-d_1)$

Delta = Delta Bare * Notional * F * 0.01

Gamma

Gamma Bare = $e^{-rT_p} * \frac{n(d_1)}{F*\sigma\sqrt{T_e}}$

Gamma = Gamma Bare * Notional * $(F * .01)^2$

Vega

Vega Bare = $e^{-rT_p} * F * n(d_1) * \sqrt{T_e}$

Vega = Vega Bare * Notional * 0.01

Rho

Rho Bare = $-T_p * e^{-rT_p} \{K * N(-d_2) + F * N(-d_1)\}$

Rhog Bare = $-T_e * e^{-rT_p} * F * N(-d_1)$

Rho = (Rho Bare + Rhoq Bare) * Notional * 0.01

Theta

Theta Bare = $-\sigma * F * e^{-rT_p} * \frac{n(-d_1)}{2*\sqrt{T_e}} + r * e^{-rT_p} * [KN(-d_2) - FN(-d_1)]$

Theta = Theta Bare * Notional / nDays

where nDays = number of days in a year

Variables Definition

K (Dirty Strike Price) = $Strike_{cleanprice} + Accrual_{expiry}$

$$F = \left[Cleanprice_{value} + Accrual_{value} - \sum_{i=first\ future\ coupon}^{i=last\ future\ coupon\ on/before\ expiry} C_i * df_{t_i} \right] / df_{t_e}$$

$$Accrual_{value} = Accrual\ from\ last\ cupon\ date\ to\ value\ date$$

$$Cleanprice_{value} = Clean\ price\ as\ of\ value\ date$$

$C_i = i^{th}\ Future\ Coupons$

$$df_{t_i} = Discount\ Factor\ on\ the\ i^{th}\ coupon\ payment\ date$$

$$df_{t_e} = Discount\ factor\ as\ of\ expiry\ date$$

3.2 American Style Bond Options

PricerBondOption uses the binomial model for American style options, which is a numerical method based on discrete time steps, configurable through the pricing parameter NUMBER_OF_TIME_STEPS. The input volatility is assumed to be the volatility of the underlying spot price, which is the bond price for bond options.

As a result, the risk measures, DELTA, GAMMA, etc. are based on price change instead of yield change. PricerBondOption does not compute an Option Adjusted Spread.

Section 4. Complex Pricers

4.1 Single Index TARN – PricerBondTarnLGMM

A target redemption note (TARN) is one in which the bond is automatically redeemed early if the sum of the coupons is above a certain target level. When the sum of the coupons is greater than the target level, the last coupon is usually reduced so that he coupon total is precisely the target.

EXSP Terms and Conditions Example Setup

Summary

Coupons:

07/11/2005 aTeaserRate

15/11/2009 max(aCap-qEURCMS10Y, 0.00)

Capital: 07/11/2005 sIN

Redemptions:

07/11/2005 ((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))

15/05/2017 ((sACI>aTargetLevel*sIN)?(xcoupon((aTargetLevel*sIN)-sAC)):(0.00))

Flow Preview

Generate

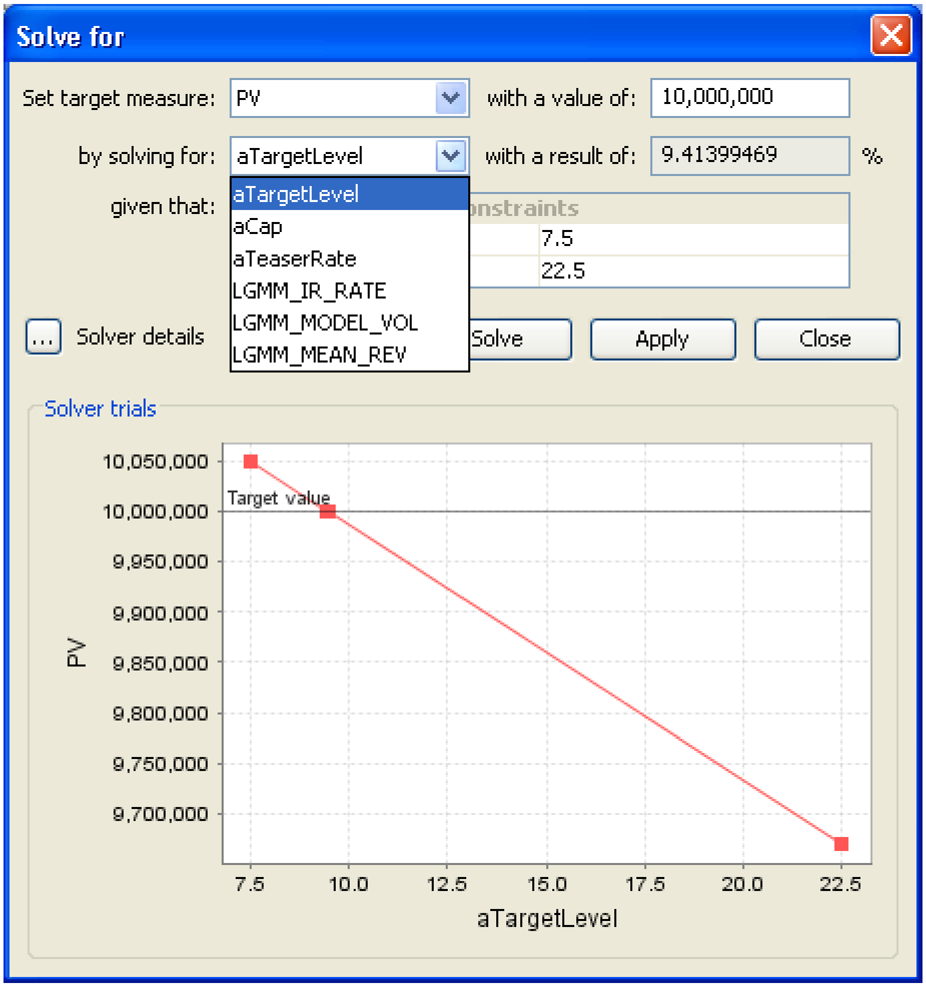
Cash Flow Date	Coupon Formula	Exotic Capital	Redemption Formula	Interest Amt	
07/11/2005				(10,000,000.00)	
15/11/2005	aTeaserRate	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	11,111.11	1
15/05/2006	aTeaserRate	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	251,388.89	1
15/11/2006	aTeaserRate	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	255,555.56	1
15/05/2007	aTeaserRate	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	251,388.89	1
15/11/2007	aTeaserRate	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	255,555.56	1
15/05/2008	aTeaserRate	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	252,777.78	1
17/11/2008	aTeaserRate	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	255,555.56	1
15/05/2009	aTeaserRate	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	251,388.89	1
16/11/2009	aTeaserRate	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	255,555.56	1
17/05/2010	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/11/2010	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
16/05/2011	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/11/2011	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/05/2012	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/11/2012	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/05/2013	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/11/2013	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/05/2014	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
17/11/2014	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/05/2015	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
16/11/2015	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
16/05/2016	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/11/2016	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
15/05/2017	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xprincipal(sIN)+xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	0.00	1
07/11/2017	max(aCap-qEURCMS10Y, 0.00)	sIN	((sACI>aTargetLevel*sIN)?(xcoupon((aTargetLevel*sIN)-sAC)):(0.00))	(1,933,536.32)	1
07/11/2017				10,000,000.00	

Notes

The Xprincipal() function is not needed in the redemption formula on the last period, the principal will be retuned automatically.

Solver

Typically, characteristics of the TARN note such as target level, initial teaser rate and constants within the coupon formula are setup as array variables. This enables them to be identified by the solver.



4.1.1 Valuation Methodology

The LGM model is used. It is the one-factor Hull-White model, expressed in HJM terms. The valuation routine is using a Monte Carlo valuation scheme. The model is calibrated to ATM caplets/swaptions corresponding to the index on the coupon.

4.1.2 Pricing Parameters

Pricing Parameter	Type	Description	Typical Value
LGMM_MEAN_REV	Rate	A transient override for the mean reversion parameter.	-1% to 5%
LGMM_MODEL_VOL	Rate	A transient override for the model's volatility parameter.	1%

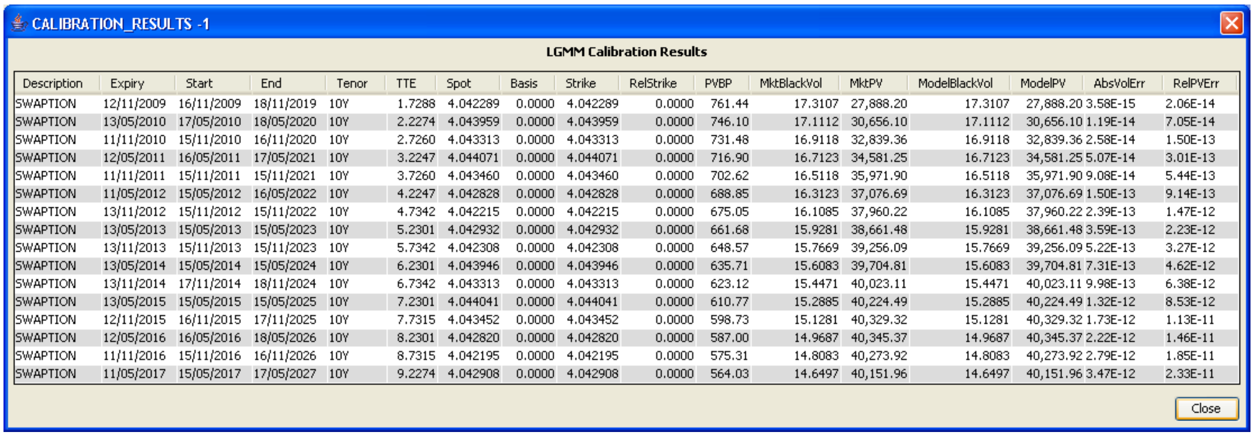
Pricing Parameter	Type	Description	Typical Value
LGMM_IR_RATE	Rate	A transient override for the yield curve.	1%-6%
LGMM_CALIBRATION_SCHEME	Choice	EXACT_STEP_SIGMA – the model volatility function is a step function, chosen so as to match the calibration instruments exactly. BEST_FIT_LM – the model mean reversion and volatility are constant and chosen by a Levenberg-Marquardt best fit routine applied to the calibration instruments. APPROX_STEP_SIGMA – same as EXACT_STEP_SIGMA except and using a faster but approximate method.	EXACT_STEP_SIGMA
NUMBER_SIMULATIONS	Integer	The number of simulations to use in the valuation routine.	33000
BROWNIAN_BRIDGE	Boolean	Controls if the path generator uses a Brownian bridge construction. Implicitly the generator will use a sobol sequence random number generator when Brownian bridge is set to TRUE. In the case of FALSE, the pseudo-random number generator is used and the path generation is classic Euler incremental generation.	TRUE
ANTITHETIC_VARIATE	Boolean	When constructs path should the antithetic path also be generated, to help reduce variance.	FALSE
LGMM_CONTROL_VARIATE	Boolean	When pricing the Bermudan, also price the first European numerically and use it as a control variates.	FALSE

Pricing Parameter	Type	Description	Typical Value
RISK_OPTIMISE	Boolean	Relates to optimization available in scenario risk report. This flag controls whether or not the optimization is used. See Pricer.getRiskExposure().	TRUE

4.1.3 Pricer Measures

CALIBRATION_RESULTS

Shows the interim results from the calibration of the LGM model.



Description	Expiry	Start	End	Tenor	TTE	Spot	Basis	Strike	RelStrike	PVBP	MktBlackVol	MktPV	ModelBlackVol	ModelPV	AbsVolErr	RelPVErr
SWAPTION	12/11/2009	16/11/2009	18/11/2019	10Y	1.7288	4.042289	0.0000	4.042289	0.0000	761.44	17.3107	27,888.20	17.3107	27,888.20	3.58E-15	2.06E-14
SWAPTION	13/05/2010	17/05/2010	18/05/2020	10Y	2.2274	4.043959	0.0000	4.043959	0.0000	746.10	17.1112	30,656.10	17.1112	30,656.10	1.19E-14	7.05E-14
SWAPTION	11/11/2010	15/11/2010	16/11/2020	10Y	2.7260	4.043313	0.0000	4.043313	0.0000	731.48	16.9118	32,839.36	16.9118	32,839.36	2.58E-14	1.50E-13
SWAPTION	12/05/2011	16/05/2011	17/05/2021	10Y	3.2247	4.044071	0.0000	4.044071	0.0000	716.90	16.7123	34,581.25	16.7123	34,581.25	5.07E-14	3.01E-13
SWAPTION	11/11/2011	15/11/2011	15/11/2021	10Y	3.7260	4.043460	0.0000	4.043460	0.0000	702.62	16.5118	35,971.90	16.5118	35,971.90	9.08E-14	5.44E-13
SWAPTION	11/05/2012	15/05/2012	16/05/2022	10Y	4.2247	4.042828	0.0000	4.042828	0.0000	688.85	16.3123	37,076.69	16.3123	37,076.69	1.50E-13	9.14E-13
SWAPTION	13/11/2012	15/11/2012	15/11/2022	10Y	4.7342	4.042215	0.0000	4.042215	0.0000	675.05	16.1085	37,960.22	16.1085	37,960.22	2.39E-13	1.47E-12
SWAPTION	13/05/2013	15/05/2013	15/05/2023	10Y	5.2301	4.042932	0.0000	4.042932	0.0000	661.68	15.9281	38,661.48	15.9281	38,661.48	3.59E-13	2.23E-12
SWAPTION	13/11/2013	15/11/2013	15/11/2023	10Y	5.7342	4.042308	0.0000	4.042308	0.0000	648.57	15.7669	39,256.09	15.7669	39,256.09	5.22E-13	3.27E-12
SWAPTION	13/05/2014	15/05/2014	15/05/2024	10Y	6.2301	4.043946	0.0000	4.043946	0.0000	635.71	15.6083	39,704.81	15.6083	39,704.81	7.31E-13	4.62E-12
SWAPTION	13/11/2014	17/11/2014	18/11/2024	10Y	6.7342	4.043313	0.0000	4.043313	0.0000	623.12	15.4471	40,023.11	15.4471	40,023.11	9.98E-13	6.38E-12
SWAPTION	13/05/2015	15/05/2015	15/05/2025	10Y	7.2301	4.044041	0.0000	4.044041	0.0000	610.77	15.2885	40,224.49	15.2885	40,224.49	1.32E-12	8.53E-12
SWAPTION	12/11/2015	16/11/2015	17/11/2025	10Y	7.7315	4.043452	0.0000	4.043452	0.0000	598.73	15.1281	40,329.32	15.1281	40,329.32	1.73E-12	1.13E-11
SWAPTION	12/05/2016	16/05/2016	18/05/2026	10Y	8.2301	4.042820	0.0000	4.042820	0.0000	587.00	14.9687	40,345.37	14.9687	40,345.37	2.22E-12	1.46E-11
SWAPTION	11/11/2016	15/11/2016	16/11/2026	10Y	8.7315	4.042195	0.0000	4.042195	0.0000	575.31	14.8083	40,273.92	14.8083	40,273.92	2.79E-12	1.85E-11
SWAPTION	11/05/2017	15/05/2017	17/05/2027	10Y	9.2274	4.042908	0.0000	4.042908	0.0000	564.03	14.6497	40,151.96	14.6497	40,151.96	3.47E-12	2.33E-11

CALIBRATION_TIME_MS

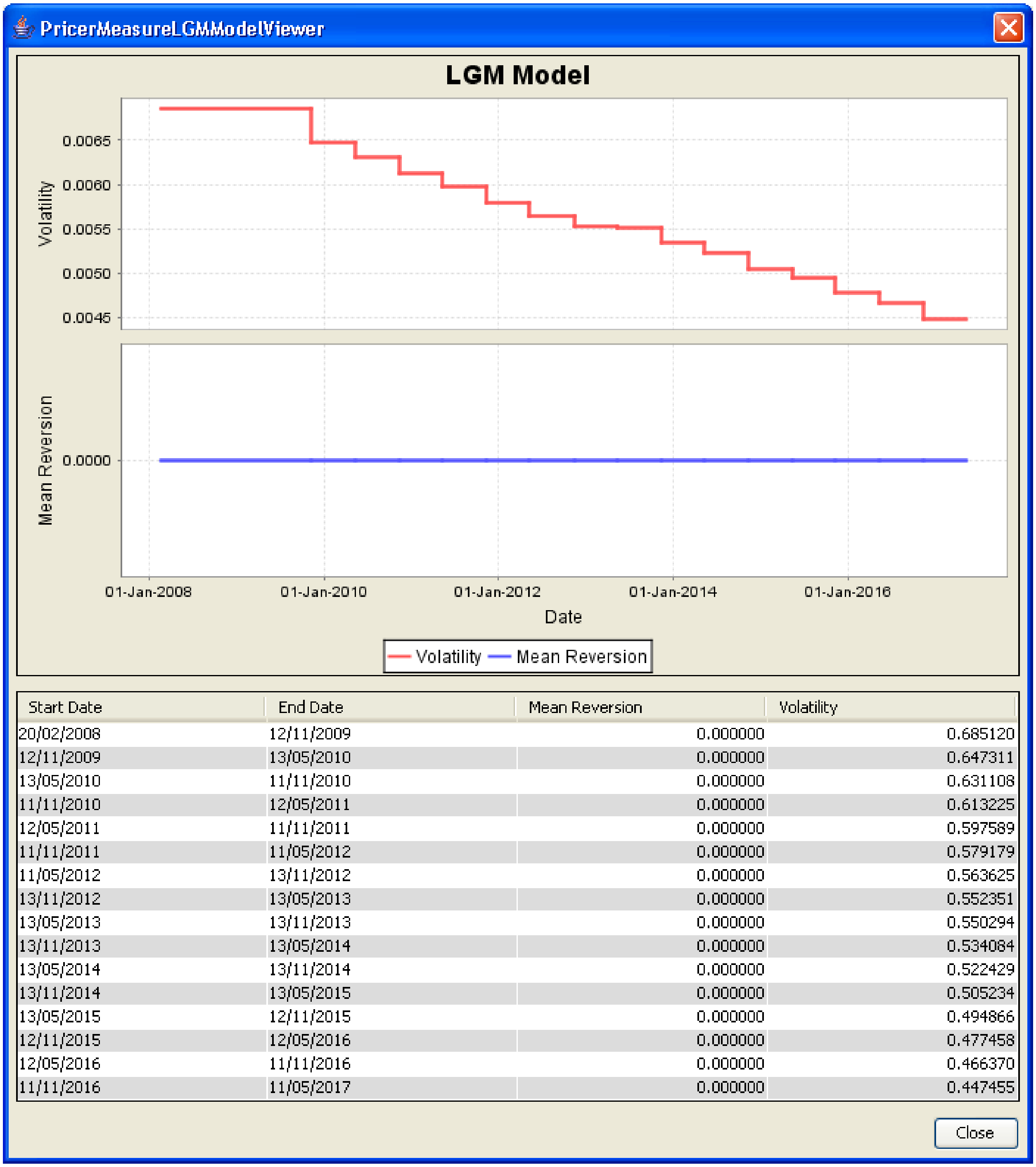
The time taken, in milliseconds, by the analytics routine to calibrate the model.

EXPECTED_TIME_TO_REDEMPTION

The expected time until redemption of the note, given the assumptions of the model, and the current pricing parameters.

LGM_MODEL

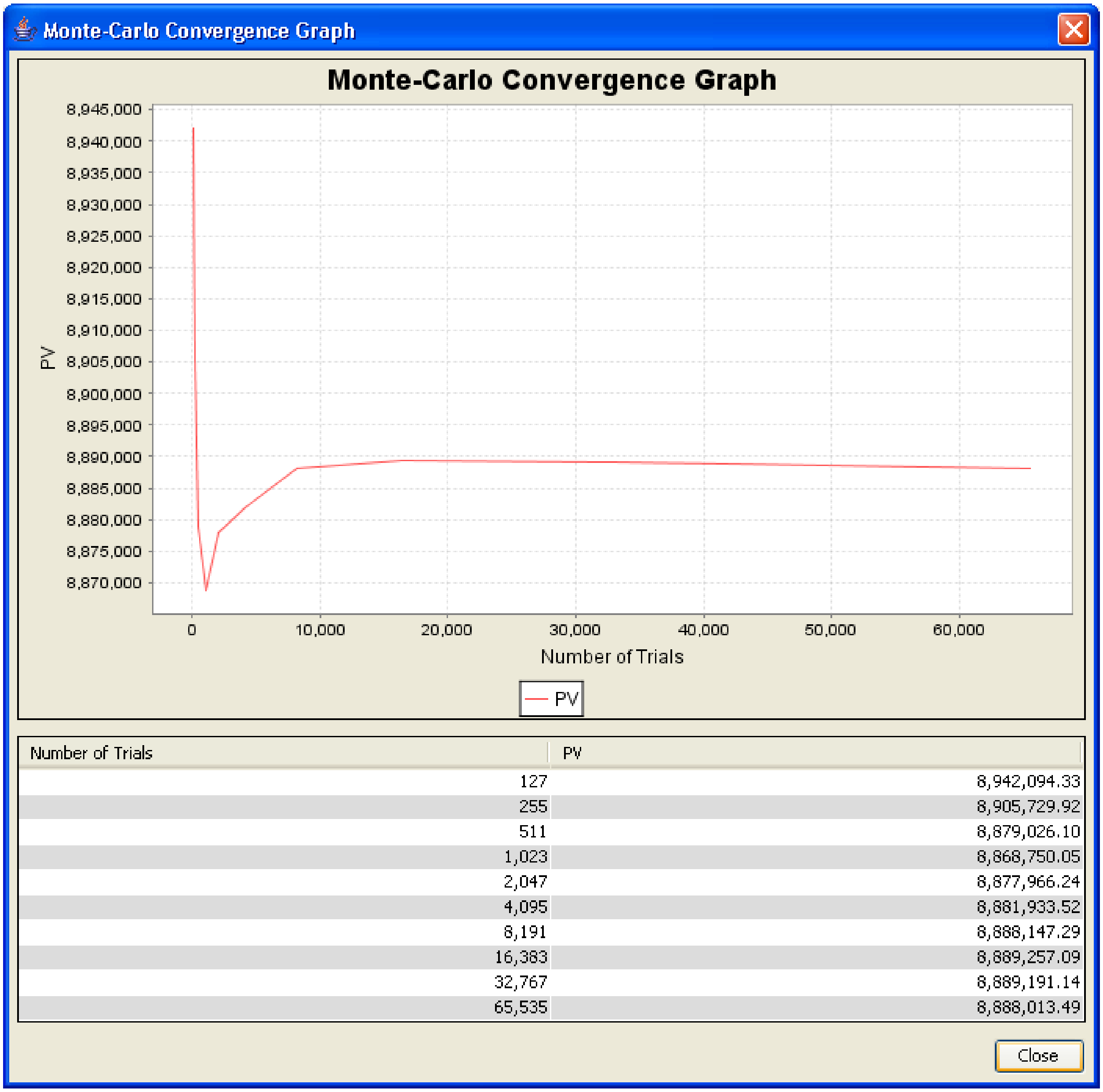
The calibrated model used to value the trade.



Start Date	End Date	Mean Reversion	Volatility
20/02/2008	12/11/2009	0.000000	0.685120
12/11/2009	13/05/2010	0.000000	0.647311
13/05/2010	11/11/2010	0.000000	0.631108
11/11/2010	12/05/2011	0.000000	0.613225
12/05/2011	11/11/2011	0.000000	0.597589
11/11/2011	11/05/2012	0.000000	0.579179
11/05/2012	13/11/2012	0.000000	0.563625
13/11/2012	13/05/2013	0.000000	0.552351
13/05/2013	13/11/2013	0.000000	0.550294
13/11/2013	13/05/2014	0.000000	0.534084
13/05/2014	13/11/2014	0.000000	0.522429
13/11/2014	13/05/2015	0.000000	0.505234
13/05/2015	12/11/2015	0.000000	0.494866
12/11/2015	12/05/2016	0.000000	0.477458
12/05/2016	11/11/2016	0.000000	0.466370
11/11/2016	11/05/2017	0.000000	0.447455

MC_GRAPH

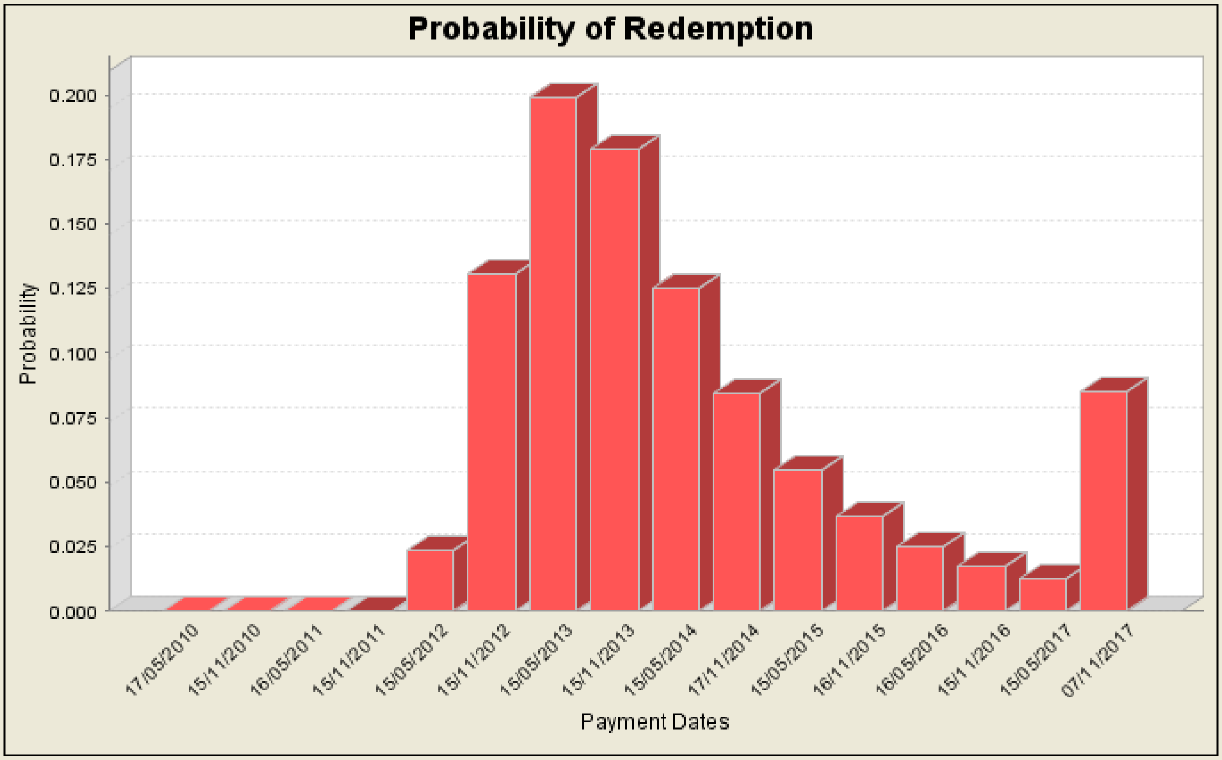
Records and displays the convergence of the Monte Carlo valuation scheme. Specifically records the NPV at discrete snapshots in the total number of simulations.



Number of Trials	PV
127	8,942,094.33
255	8,905,729.92
511	8,879,026.10
1,023	8,868,750.05
2,047	8,877,966.24
4,095	8,881,933.52
8,191	8,888,147.29
16,383	8,889,257.09
32,767	8,889,191.14
65,535	8,888,013.49

PROB_OF_REDEMPTION

Displays the probability of redemption of the bond at each coupon payment date, given that the bond has not been redeemed already.



Payment Dates	Probability
17/05/2010	0.0000
15/11/2010	0.0000
16/05/2011	0.0000
15/11/2011	0.0076
15/05/2012	2.4094
15/11/2012	13.4356
15/05/2013	20.4524
15/11/2013	18.3780
15/05/2014	12.8626
17/11/2014	8.6458
15/05/2015	5.6451
16/11/2015	3.7438
16/05/2016	2.5635
15/11/2016	1.8257
15/05/2017	1.2871
07/11/2017	8.7434

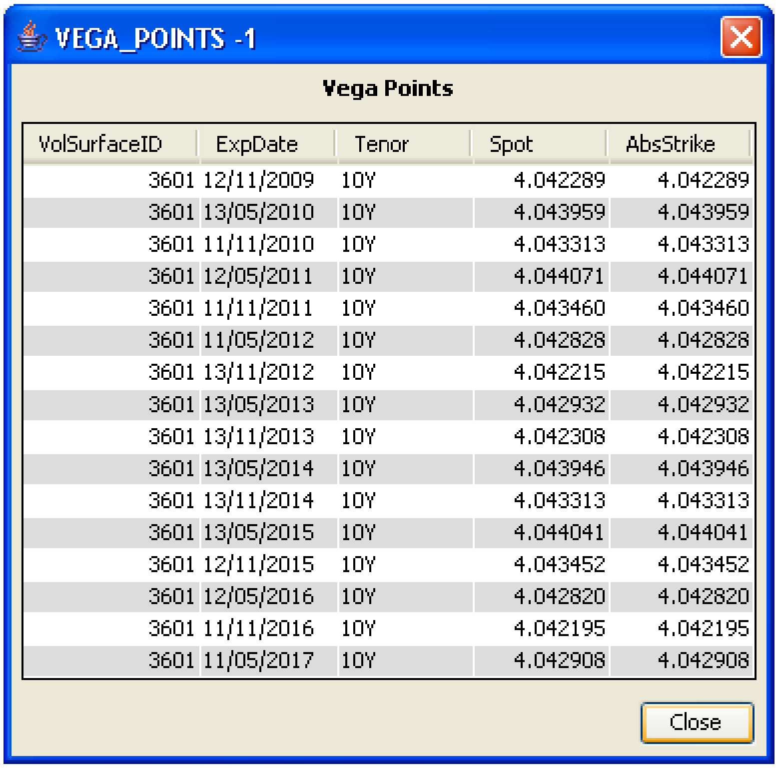
Close

VALUATION_TIME_MS

The time taken by the core valuation routine to compute the requested measures. It explicitly does not include the model calibration time.

VEGA_POINTS

Records which points on the volatility surface have been used in valuation. These are typically used in optimization of vega reports.



VolSurfaceID	ExpDate	Tenor	Spot	AbsStrike
3601	12/11/2009	10Y	4.042289	4.042289
3601	13/05/2010	10Y	4.043959	4.043959
3601	11/11/2010	10Y	4.043313	4.043313
3601	12/05/2011	10Y	4.044071	4.044071
3601	11/11/2011	10Y	4.043460	4.043460
3601	11/05/2012	10Y	4.042828	4.042828
3601	13/11/2012	10Y	4.042215	4.042215
3601	13/05/2013	10Y	4.042932	4.042932
3601	13/11/2013	10Y	4.042308	4.042308
3601	13/05/2014	10Y	4.043946	4.043946
3601	13/11/2014	10Y	4.043313	4.043313
3601	13/05/2015	10Y	4.044041	4.044041
3601	12/11/2015	10Y	4.043452	4.043452
3601	12/05/2016	10Y	4.042820	4.042820
3601	11/11/2016	10Y	4.042195	4.042195
3601	11/05/2017	10Y	4.042908	4.042908

4.2 Multi-Index TARN – PricerBondLGMM2F

In the case of a multi-index TARN, the coupon is a function of more than one index, typically, a leveraged CMS spread index.

EXSP Terms and Conditions Example Setup

Summary

Coupons:

31/01/2008 aTeaserRate

15/01/2009 aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))

Capital: 31/01/2008 sIN

Redemptions:

31/01/2008 ((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))

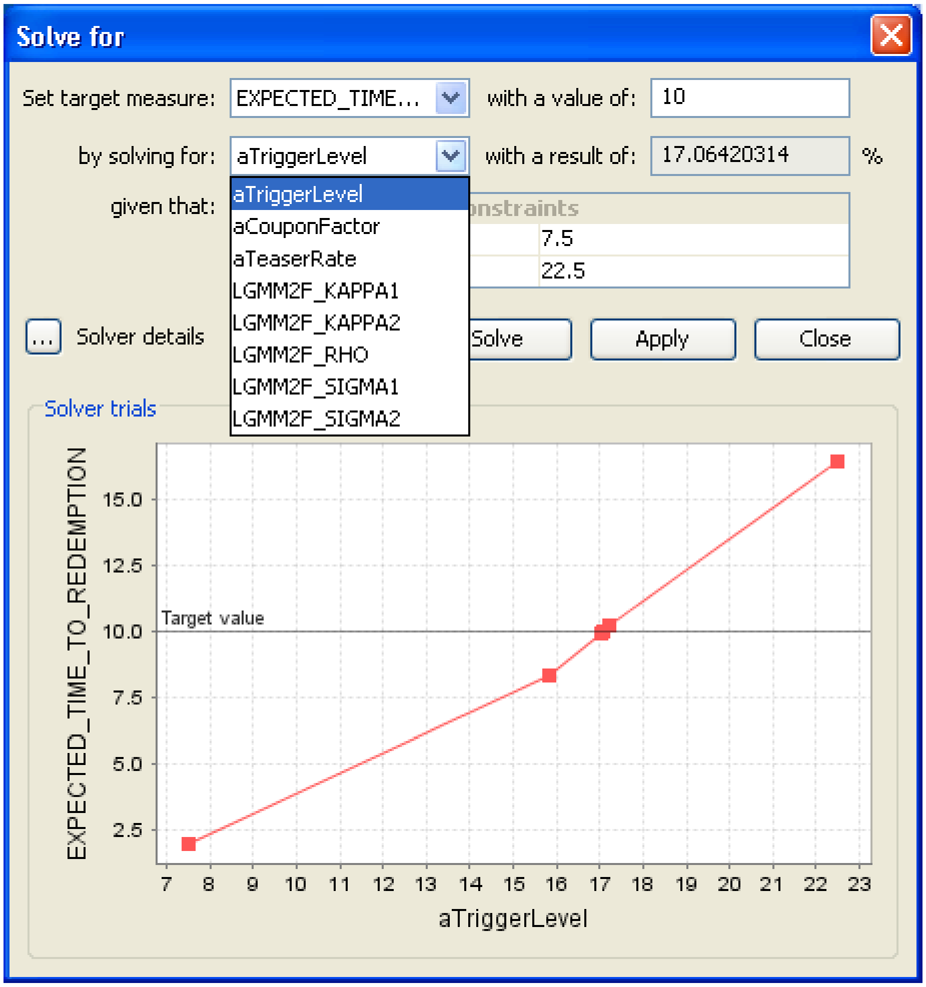
15/01/2038 ((sACI>sIN*aTriggerLevel)?(xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))

Generate

Cash Flow Date	Coupon Formula	Exotic Capital	Redemption Formula	Interest Amt	
31/01/2008				(1,000,000.00)	
15/07/2008	aTeaserRate	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	32,277.78	1
15/01/2009	aTeaserRate	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	35,777.78	1
15/07/2009	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
15/01/2010	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
15/07/2010	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
18/01/2011	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
15/07/2011	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
17/01/2012	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
16/07/2012	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
15/01/2013	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
15/07/2013	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
15/01/2014	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	1
15/07/2033	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
17/01/2034	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
17/07/2034	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
16/01/2035	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
16/07/2035	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
15/01/2036	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
15/07/2036	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
15/01/2037	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
15/07/2037	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
15/01/2038	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xprincipal(sIN)+xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	0.00	
01/02/2038	aCouponFactor*(max(qUSDCMS30Y-qUSDLIBOR6M, 0.00))	sIN	((sACI>sIN*aTriggerLevel)?(xcoupon((aTriggerLevel*sIN)-sAC)):(0.00))	(4,173,202.53)	
01/02/2038				1,000,000.00	

Solver

Typically, one sets up the teaser rate, the coupon constants and the trigger level, as an array variable, in which case one can solve for it given one of the target pricer measures.



4.2.1 Valuation Methodology

The LGM2F model is used. It is the two-factor Hull-White model, expressed in HJM terms. The valuation routine is using a Monte Carlo valuation scheme. The model is calibrated to ATM caplets/swaptions corresponding to the indexes on the coupon.

4.2.2 Pricing Parameters

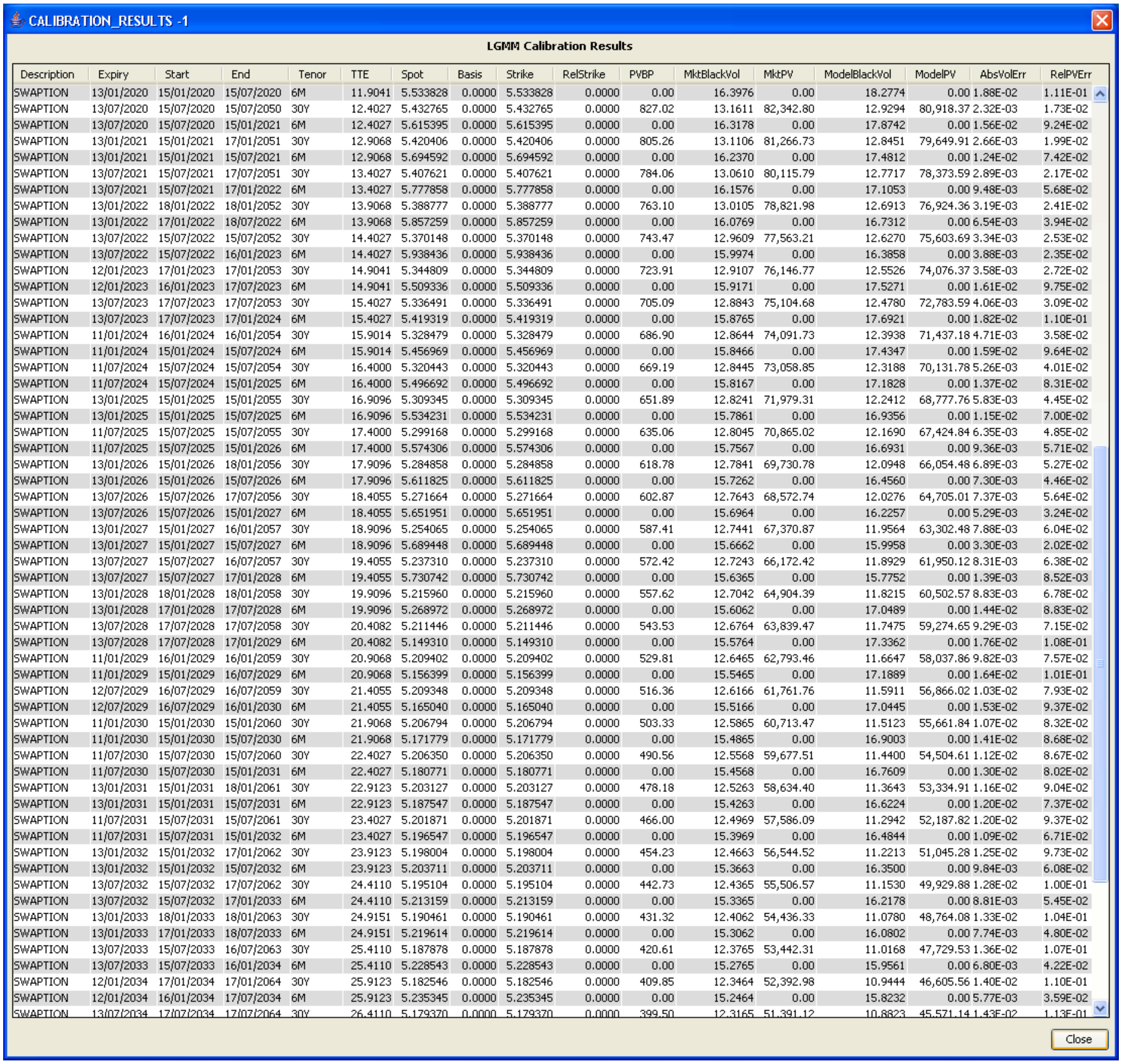
Pricing Parameter	Type	Description	Typical Value
LGMM2F_KAPPA1	Rate	A transient override for the first mean reversion parameter.	-1% to 5%
LGMM2F_KAPPA2	Rate	A transient override for the second mean reversion parameter.	20%-120%
LGMM2F_SIGMA1	Rate	A transient override for the first volatility parameter.	0.5%-2%

Pricing Parameter	Type	Description	Typical Value
LGMM2F_SIGMA2	Rate	A transient override for the second volatility parameter.	0.5%-2%
LGMM2F_RHO	Rate	A transient override for the correlation parameter.	-100%-100%
IGNORE_TARN	Boolean	The convenient transient flag to allow the valuation scheme to ignore the TARN feature, and so compare pricing with and without TARN.	FALSE
NUMBER_SIMULATIONS	Integer	The number of simulations to use in the valuation routine.	33000
BROWNIAN_BRIDGE	Boolean	Controls if the path generator uses a Brownian bridge construction. Implicitly the generator will use a sobol sequence random number generator when Brownian bridge is set to TRUE. In the case of FALSE, the pseudo-random number generator is used and the path generation is classic Euler incremental generation.	TRUE
RISK_OPTIMISE	Boolean	Relates to optimization available in scenario risk report. This flag controls whether or not the optimization is used. See Pricer.getRiskExposure().	TRUE

4.2.3 Pricer Measures

CALIBRATION_RESULTS

Shows the interim results from the calibration of the LGM2F model.



Description	Expiry	Start	End	Tenor	TTE	Spot	Basis	Strike	RelStrike	PVBP	MktBlackVol	MktPV	ModelBlackVol	ModelPV	AbsVolErr	RelPVErr
SWAPTION	13/01/2020	15/01/2020	15/07/2020	6M	11.9041	5.533828	0.0000	5.533828	0.0000	0.00	16.3976	0.00	18.2774	0.00	1.88E-02	1.11E-01
SWAPTION	13/07/2020	15/07/2020	15/07/2050	30Y	12.4027	5.432765	0.0000	5.432765	0.0000	827.02	13.1611	82,342.80	12.9294	80,918.37	2.32E-03	1.73E-02
SWAPTION	13/07/2020	15/07/2020	15/01/2021	6M	12.4027	5.615395	0.0000	5.615395	0.0000	0.00	16.3178	0.00	17.8742	0.00	1.56E-02	9.24E-02
SWAPTION	13/01/2021	15/01/2021	17/01/2051	30Y	12.9068	5.420406	0.0000	5.420406	0.0000	805.26	13.1106	81,266.73	12.8451	79,649.91	2.66E-03	1.99E-02
SWAPTION	13/01/2021	15/01/2021	15/07/2021	6M	12.9068	5.694592	0.0000	5.694592	0.0000	0.00	16.2370	0.00	17.4812	0.00	1.24E-02	7.42E-02
SWAPTION	13/07/2021	15/07/2021	17/07/2051	30Y	13.4027	5.407621	0.0000	5.407621	0.0000	784.06	13.0610	80,115.79	12.7717	78,373.59	2.89E-03	2.17E-02
SWAPTION	13/07/2021	15/07/2021	17/01/2022	6M	13.4027	5.777858	0.0000	5.777858	0.0000	0.00	16.1576	0.00	17.1053	0.00	9.48E-03	5.68E-02
SWAPTION	13/01/2022	18/01/2022	18/01/2052	30Y	13.9068	5.388777	0.0000	5.388777	0.0000	763.10	13.0105	78,821.98	12.6913	76,924.36	3.19E-03	2.41E-02
SWAPTION	13/01/2022	17/01/2022	18/07/2022	6M	13.9068	5.857259	0.0000	5.857259	0.0000	0.00	16.0769	0.00	16.7312	0.00	6.54E-03	3.94E-02
SWAPTION	13/07/2022	15/07/2022	15/07/2052	30Y	14.4027	5.370148	0.0000	5.370148	0.0000	743.47	12.9609	77,563.21	12.6270	75,603.69	3.34E-03	2.53E-02
SWAPTION	13/07/2022	15/07/2022	16/01/2023	6M	14.4027	5.938436	0.0000	5.938436	0.0000	0.00	15.9974	0.00	16.3858	0.00	3.88E-03	2.35E-02
SWAPTION	12/01/2023	17/01/2023	17/01/2053	30Y	14.9041	5.344809	0.0000	5.344809	0.0000	723.91	12.9107	76,146.77	12.5526	74,076.37	3.58E-03	2.72E-02
SWAPTION	12/01/2023	16/01/2023	17/07/2023	6M	14.9041	5.509336	0.0000	5.509336	0.0000	0.00	15.9171	0.00	17.5271	0.00	1.61E-02	9.75E-02
SWAPTION	13/07/2023	17/07/2023	17/07/2053	30Y	15.4027	5.336491	0.0000	5.336491	0.0000	705.09	12.8843	75,104.68	12.4780	72,783.59	4.06E-03	3.09E-02
SWAPTION	13/07/2023	17/07/2023	17/01/2024	6M	15.4027	5.419319	0.0000	5.419319	0.0000	0.00	15.8765	0.00	17.6921	0.00	1.82E-02	1.10E-01
SWAPTION	11/01/2024	16/01/2024	16/01/2054	30Y	15.9014	5.328479	0.0000	5.328479	0.0000	686.90	12.8644	74,091.73	12.3938	71,437.18	4.71E-03	3.58E-02
SWAPTION	11/01/2024	15/01/2024	15/07/2024	6M	15.9014	5.456969	0.0000	5.456969	0.0000	0.00	15.8466	0.00	17.4347	0.00	1.59E-02	9.64E-02
SWAPTION	11/07/2024	15/07/2024	15/07/2054	30Y	16.4000	5.320443	0.0000	5.320443	0.0000	669.19	12.8445	73,058.85	12.3188	70,131.78	5.26E-03	4.01E-02
SWAPTION	11/07/2024	15/07/2024	15/01/2025	6M	16.4000	5.496692	0.0000	5.496692	0.0000	0.00	15.8167	0.00	17.1828	0.00	1.37E-02	8.31E-02
SWAPTION	13/01/2025	15/01/2025	15/01/2055	30Y	16.9096	5.309345	0.0000	5.309345	0.0000	651.89	12.8241	71,979.31	12.2412	68,777.76	5.83E-03	4.45E-02
SWAPTION	13/01/2025	15/01/2025	15/07/2025	6M	16.9096	5.534231	0.0000	5.534231	0.0000	0.00	15.7861	0.00	16.9356	0.00	1.15E-02	7.00E-02
SWAPTION	11/07/2025	15/07/2025	15/07/2055	30Y	17.4000	5.299168	0.0000	5.299168	0.0000	635.06	12.8045	70,865.02	12.1690	67,424.84	6.35E-03	4.85E-02
SWAPTION	11/07/2025	15/07/2025	15/01/2026	6M	17.4000	5.574306	0.0000	5.574306	0.0000	0.00	15.7567	0.00	16.6931	0.00	9.36E-03	5.71E-02
SWAPTION	13/01/2026	15/01/2026	18/01/2056	30Y	17.9096	5.284858	0.0000	5.284858	0.0000	618.78	12.7841	69,730.78	12.0948	66,054.48	6.89E-03	5.27E-02
SWAPTION	13/01/2026	15/01/2026	15/07/2026	6M	17.9096	5.611825	0.0000	5.611825	0.0000	0.00	15.7262	0.00	16.4560	0.00	7.30E-03	4.46E-02
SWAPTION	13/07/2026	15/07/2026	17/07/2056	30Y	18.4055	5.271664	0.0000	5.271664	0.0000	602.87	12.7643	68,572.74	12.0276	64,705.01	7.37E-03	5.64E-02
SWAPTION	13/07/2026	15/07/2026	15/01/2027	6M	18.4055	5.651951	0.0000	5.651951	0.0000	0.00	15.6964	0.00	16.2257	0.00	5.29E-03	3.24E-02
SWAPTION	13/01/2027	15/01/2027	16/01/2057	30Y	18.9096	5.254065	0.0000	5.254065	0.0000	587.41	12.7441	67,370.87	11.9564	63,302.48	7.88E-03	6.04E-02
SWAPTION	13/01/2027	15/01/2027	15/07/2027	6M	18.9096	5.689448	0.0000	5.689448	0.0000	0.00	15.6662	0.00	15.9958	0.00	3.30E-03	2.02E-02
SWAPTION	13/07/2027	15/07/2027	16/07/2057	30Y	19.4055	5.237310	0.0000	5.237310	0.0000	572.42	12.7243	66,172.42	11.8929	61,950.12	8.31E-03	6.38E-02
SWAPTION	13/07/2027	15/07/2027	17/01/2028	6M	19.4055	5.730742	0.0000	5.730742	0.0000	0.00	15.6365	0.00	15.7752	0.00	1.39E-03	8.52E-03
SWAPTION	13/01/2028	18/01/2028	18/01/2058	30Y	19.9096	5.215960	0.0000	5.215960	0.0000	557.62	12.7042	64,904.39	11.8215	60,502.57	8.83E-03	6.78E-02
SWAPTION	13/01/2028	17/01/2028	17/07/2028	6M	19.9096	5.268972	0.0000	5.268972	0.0000	0.00	15.6062	0.00	17.0489	0.00	1.44E-02	8.83E-02
SWAPTION	13/07/2028	17/07/2028	17/07/2058	30Y	20.4082	5.211446	0.0000	5.211446	0.0000	543.53	12.6764	63,839.47	11.7475	59,274.65	9.29E-03	7.15E-02
SWAPTION	13/07/2028	17/07/2028	17/01/2029	6M	20.4082	5.149310	0.0000	5.149310	0.0000	0.00	15.5764	0.00	17.3362	0.00	1.76E-02	1.08E-01
SWAPTION	11/01/2029	16/01/2029	16/01/2059	30Y	20.9068	5.209402	0.0000	5.209402	0.0000	529.81	12.6465	62,793.46	11.6647	58,037.86	9.82E-03	7.57E-02
SWAPTION	11/01/2029	15/01/2029	16/07/2029	6M	20.9068	5.156399	0.0000	5.156399	0.0000	0.00	15.5465	0.00	17.1889	0.00	1.64E-02	1.01E-01
SWAPTION	12/07/2029	16/07/2029	16/07/2059	30Y	21.4055	5.209348	0.0000	5.209348	0.0000	516.36	12.6166	61,761.76	11.5911	56,866.02	1.03E-02	7.93E-02
SWAPTION	12/07/2029	16/07/2029	16/01/2030	6M	21.4055	5.165040	0.0000	5.165040	0.0000	0.00	15.5166	0.00	17.0445	0.00	1.53E-02	9.37E-02
SWAPTION	11/01/2030	15/01/2030	15/01/2060	30Y	21.9068	5.206794	0.0000	5.206794	0.0000	503.33	12.5865	60,713.47	11.5123	55,661.84	1.07E-02	8.32E-02
SWAPTION	11/01/2030	15/01/2030	15/07/2030	6M	21.9068	5.171779	0.0000	5.171779	0.0000	0.00	15.4865	0.00	16.9003	0.00	1.41E-02	8.68E-02
SWAPTION	11/07/2030	15/07/2030	15/07/2060	30Y	22.4027	5.206350	0.0000	5.206350	0.0000	490.56	12.5568	59,677.51	11.4400	54,504.61	1.12E-02	8.67E-02
SWAPTION	11/07/2030	15/07/2030	15/01/2031	6M	22.4027	5.180771	0.0000	5.180771	0.0000	0.00	15.4568	0.00	16.7609	0.00	1.30E-02	8.02E-02
SWAPTION	13/01/2031	15/01/2031	18/01/2061	30Y	22.9123	5.203127	0.0000	5.203127	0.0000	478.18	12.5263	58,634.40	11.3643	53,334.91	1.16E-02	9.04E-02
SWAPTION	13/01/2031	15/01/2031	15/07/2031	6M	22.9123	5.187547	0.0000	5.187547	0.0000	0.00	15.4263	0.00	16.6224	0.00	1.20E-02	7.37E-02
SWAPTION	11/07/2031	15/07/2031	15/07/2061	30Y	23.4027	5.201871	0.0000	5.201871	0.0000	466.00	12.4969	57,586.09	11.2942	52,187.82	1.20E-02	9.37E-02
SWAPTION	11/07/2031	15/07/2031	15/01/2032	6M	23.4027	5.196547	0.0000	5.196547	0.0000	0.00	15.3969	0.00	16.4844	0.00	1.09E-02	6.71E-02
SWAPTION	13/01/2032	15/01/2032	17/01/2062	30Y	23.9123	5.198004	0.0000	5.198004	0.0000	454.23	12.4663	56,544.52	11.2213	51,045.28	1.25E-02	9.73E-02
SWAPTION	13/01/2032	15/01/2032	15/07/2032	6M	23.9123	5.203711	0.0000	5.203711	0.0000	0.00	15.3663	0.00	16.3500	0.00	9.84E-03	6.08E-02
SWAPTION	13/07/2032	15/07/2032	17/07/2062	30Y	24.4110	5.195104	0.0000	5.195104	0.0000	442.73	12.4365	55,506.57	11.1530	49,929.88	1.28E-02	1.00E-01
SWAPTION	13/07/2032	15/07/2032	17/01/2033	6M	24.4110	5.213159	0.0000	5.213159	0.0000	0.00	15.3365	0.00	16.2178	0.00	8.81E-03	5.45E-02
SWAPTION	13/01/2033	18/01/2033	18/01/2063	30Y	24.9151	5.190461	0.0000	5.190461	0.0000	431.32	12.4062	54,436.33	11.0780	48,764.08	1.33E-02	1.04E-01
SWAPTION	13/01/2033	17/01/2033	18/07/2033	6M	24.9151	5.219614	0.0000	5.219614	0.0000	0.00	15.3062	0.00	16.0802	0.00	7.74E-03	4.80E-02
SWAPTION	13/07/2033	15/07/2033	16/07/2063	30Y	25.4110	5.187878	0.0000	5.187878	0.0000	420.61	12.3765	53,442.31	11.0168	47,729.53	1.36E-02	1.07E-01
SWAPTION	13/07/2033	15/07/2033	16/01/2034	6M	25.4110	5.228543	0.0000	5.228543	0.0000	0.00	15.2765	0.00	15.9561	0.00	6.80E-03	4.22E-02
SWAPTION	12/01/2034	17/01/2034	17/01/2064	30Y	25.9123	5.182546	0.0000	5.182546	0.0000	409.85	12.3464	52,392.98	10.9444	46,605.56	1.40E-02	1.10E-01
SWAPTION	12/01/2034	16/01/2034	17/07/2034	6M	25.9123	5.235345	0.0000	5.235345	0.0000	0.00	15.2464	0.00	15.8232	0.00	5.77E-03	3.59E-02
SWAPTION	13/07/2034	17/07/2034	17/07/2064	30Y	26.4110	5.179370	0.0000	5.179370	0.0000	399.50	12.3165	51,391.12	10.8823	45,571.14	1.43E-02	1.13E-01

CALIBRATION_TIME_MS

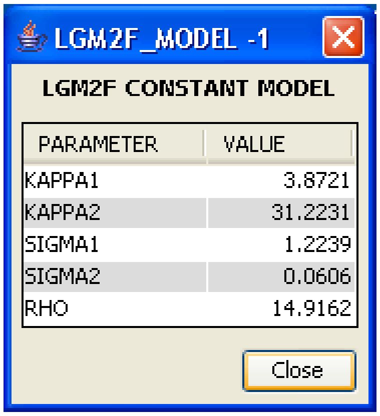
The time taken by the analytics to calibrate the model

EXPECTED_TIME_TO_REDEMPTION

The expected time until redemption of the note, given the assumptions of the model, and the current pricing parameters.

LGM2F_MODEL

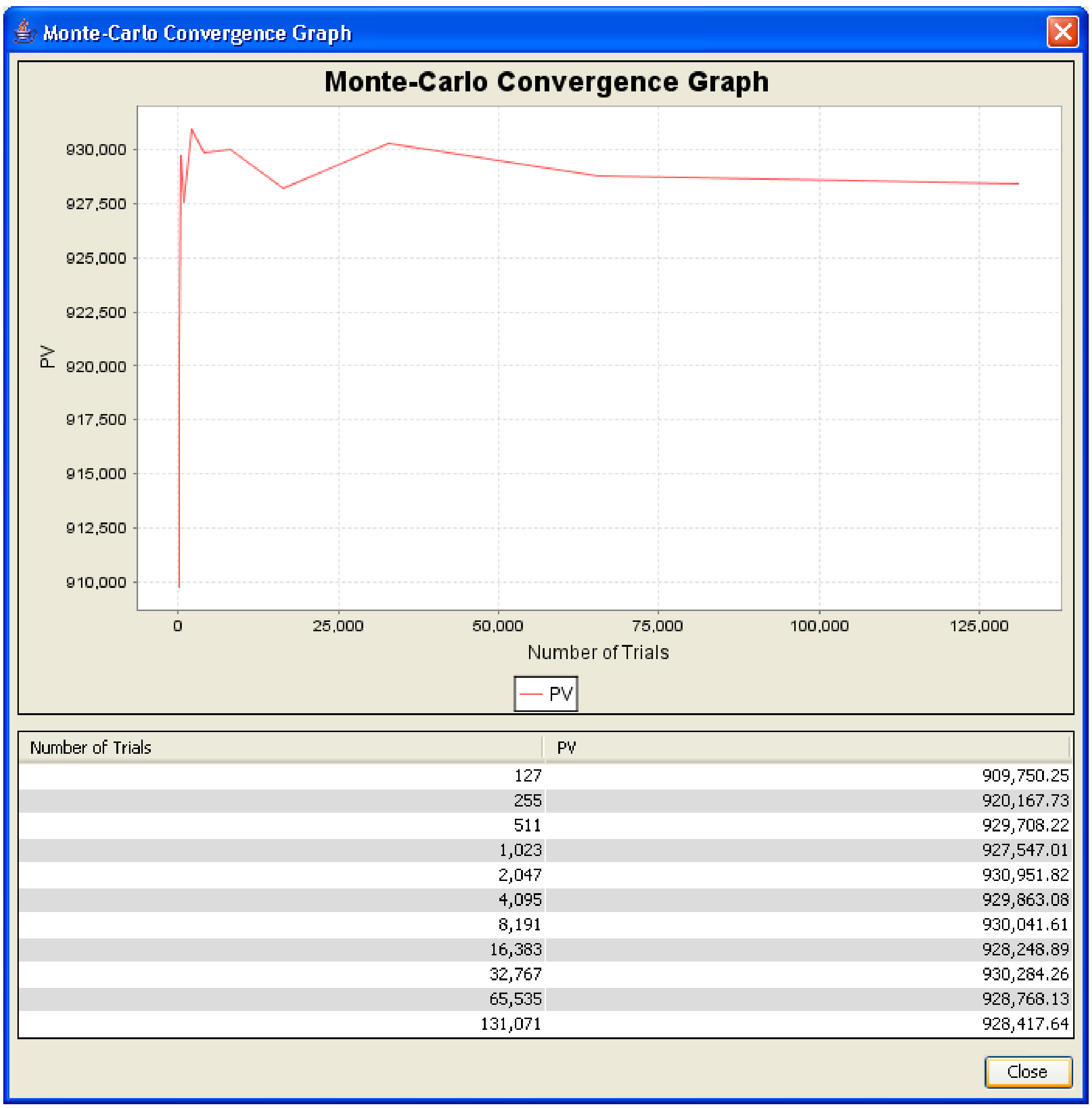
Sows the calibrated model used to value the trade.



PARAMETER	VALUE
KAPPA1	3.8721
KAPPA2	31.2231
SIGMA1	1.2239
SIGMA2	0.0606
RHO	14.9162

MC_GRAPH

The convergence graph of the Monte Carlo valuation routine.

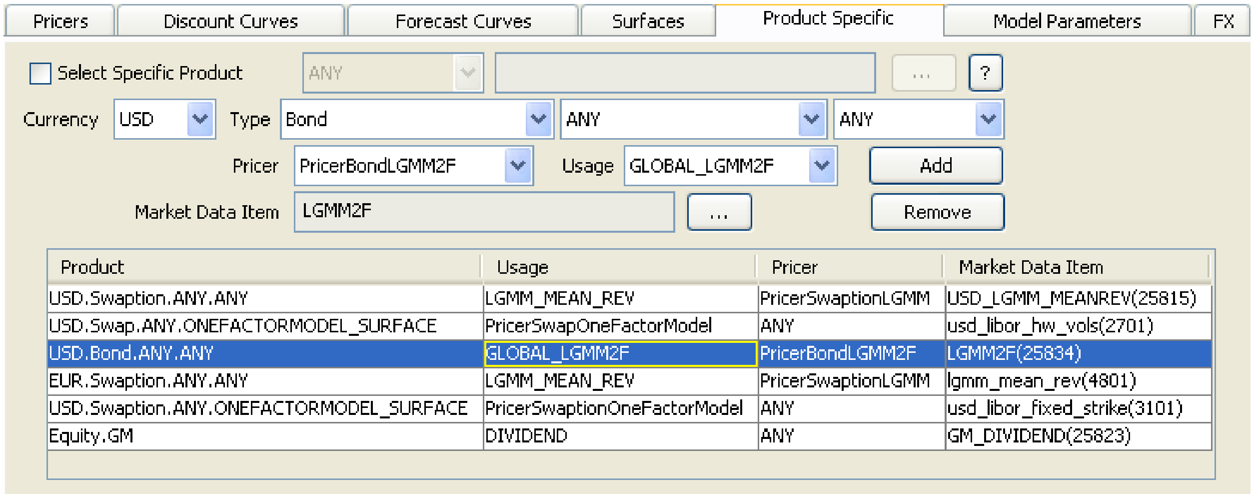


Number of Trials	PV
127	909,750.25
255	920,167.73
511	929,708.22
1,023	927,547.01
2,047	930,951.82
4,095	929,863.08
8,191	930,041.61
16,383	928,248.89
32,767	930,284.26
65,535	928,768.13
131,071	928,417.64

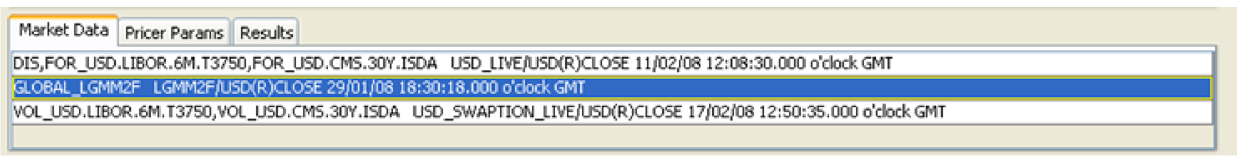
Market Data Configuration

GLOBAL_LGMM2F Usage

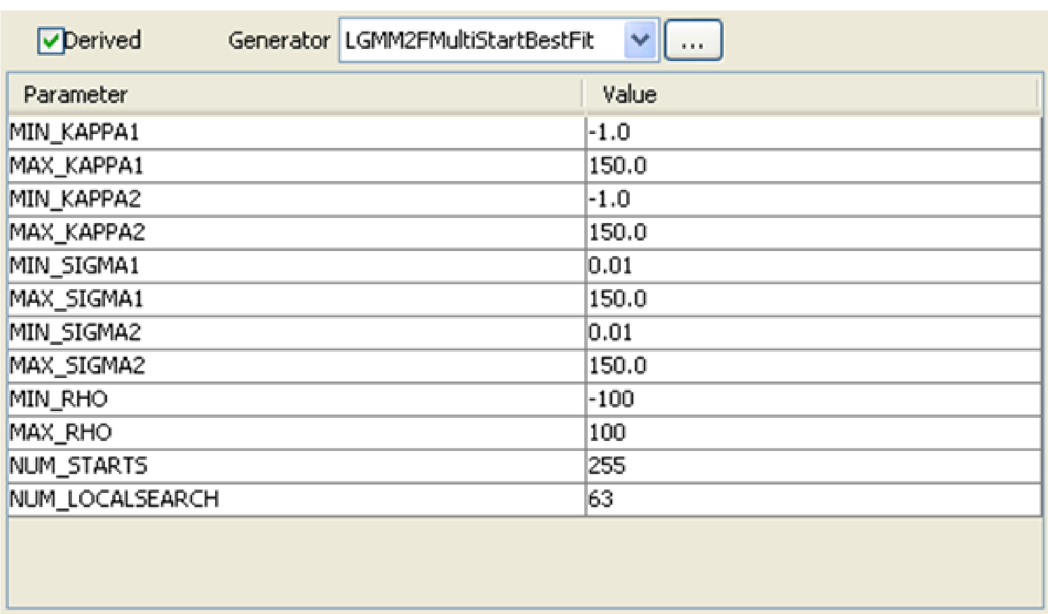
The valuation routine requires LGMM2F model parameters setup in a market data item. The assignment within the pricing config is via Product Specific panel, see below.



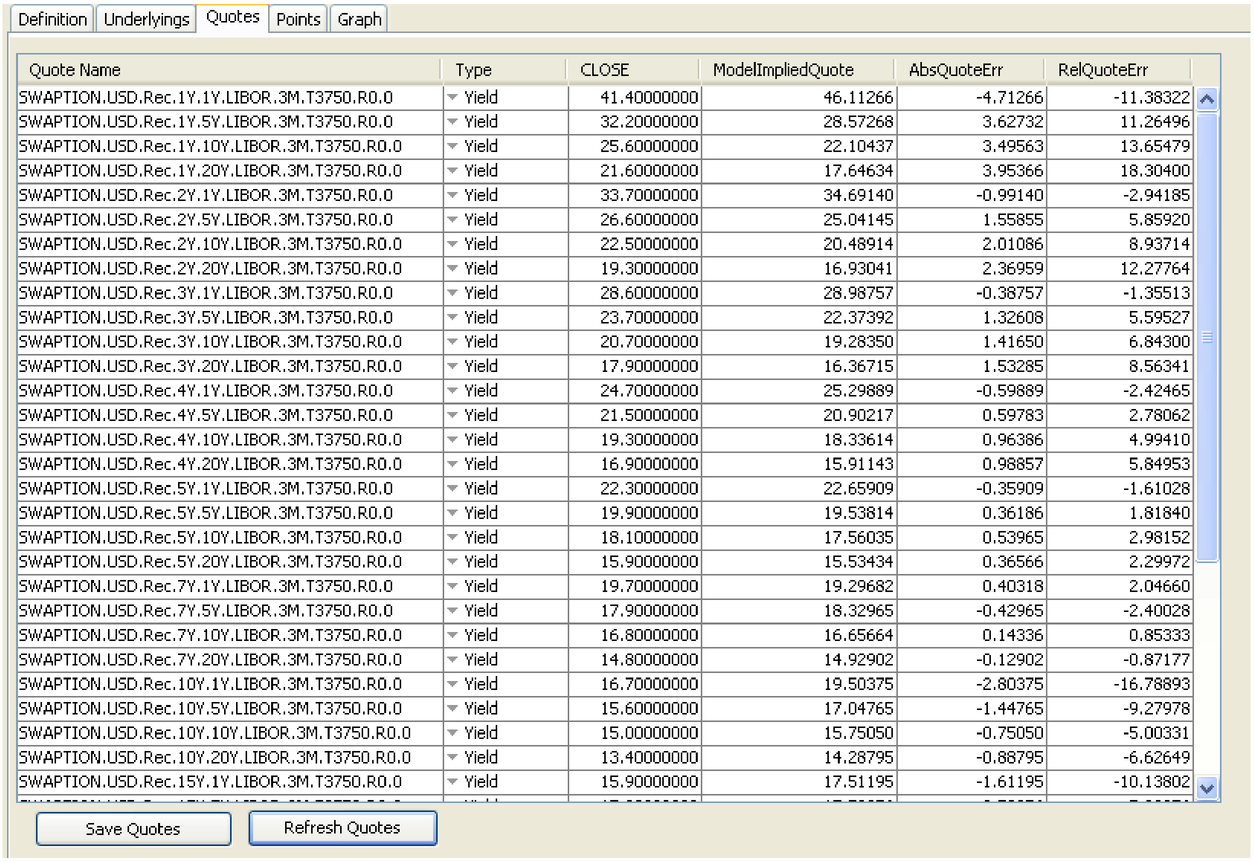
Product	Usage	Pricer	Market Data Item
USD.Swaption.ANY.ANY	LGMM_MEAN_REV	PricerSwaptionLGMM	USD_LGMM_MEANREV(25815)
USD.Swap.ANY.ONEFACTORMODEL_SURFACE	PricerSwapOneFactorModel	ANY	usd_libor_hw_vols(2701)
USD.Bond.ANY.ANY	GLOBAL_LGMM2F	PricerBondLGMM2F	LGMM2F(25834)
EUR.Swaption.ANY.ANY	LGMM_MEAN_REV	PricerSwaptionLGMM	lgmm_mean_rev(4801)
USD.Swaption.ANY.ONEFACTORMODEL_SURFACE	PricerSwaptionOneFactorModel	ANY	usd_libor_fixed_strike(3101)
Equity.GM	DIVIDEND	ANY	GM_DIVIDEND(25823)



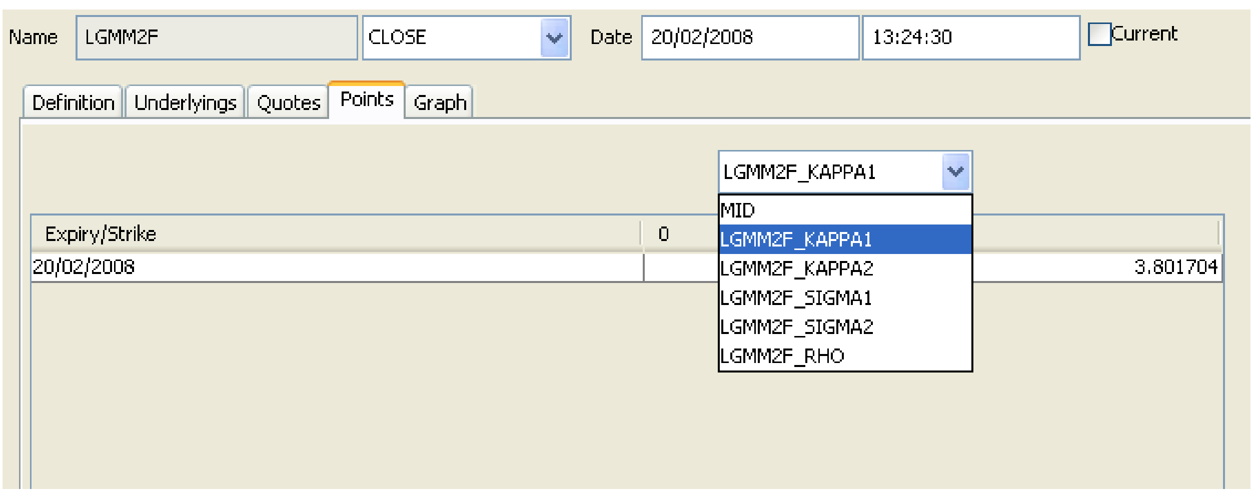
The global LGMM2F market data surface requires the generator LGMM2FMultiStartBestFit. This generator calibrates in a best fit sense ATM swaptions. Typically, one calibrates to the entire ATM vol surface. This can take several minutes, depending on the number of quotes in the surface. The solution is a set of model parameters, kappa1, kappa2, sigma1, sigma2, rho. These are then refined slightly with the trade valuation to provide a sharper fit to a subset of the ATM vols relevant to the trade.



Parameter	Value
MIN_KAPPA1	-1.0
MAX_KAPPA1	150.0
MIN_KAPPA2	-1.0
MAX_KAPPA2	150.0
MIN_SIGMA1	0.01
MAX_SIGMA1	150.0
MIN_SIGMA2	0.01
MAX_SIGMA2	150.0
MIN_RHO	-100
MAX_RHO	100
NUM_STARTS	255
NUM_LOCALSEARCH	63



Quote Name	Type	CLOSE	ModelImpliedQuote	AbsQuoteErr	RelQuoteErr
SWAPTION.USD.Rec.1Y.1Y.LIBOR.3M.T3750.R0.0	Yield	41.40000000	46.11266	-4.71266	-11.38322
SWAPTION.USD.Rec.1Y.5Y.LIBOR.3M.T3750.R0.0	Yield	32.20000000	28.57268	3.62732	11.26496
SWAPTION.USD.Rec.1Y.10Y.LIBOR.3M.T3750.R0.0	Yield	25.60000000	22.10437	3.49563	13.65479
SWAPTION.USD.Rec.1Y.20Y.LIBOR.3M.T3750.R0.0	Yield	21.60000000	17.64634	3.95366	18.30400
SWAPTION.USD.Rec.2Y.1Y.LIBOR.3M.T3750.R0.0	Yield	33.70000000	34.69140	-0.99140	-2.94185
SWAPTION.USD.Rec.2Y.5Y.LIBOR.3M.T3750.R0.0	Yield	26.60000000	25.04145	1.55855	5.85920
SWAPTION.USD.Rec.2Y.10Y.LIBOR.3M.T3750.R0.0	Yield	22.50000000	20.48914	2.01086	8.93714
SWAPTION.USD.Rec.2Y.20Y.LIBOR.3M.T3750.R0.0	Yield	19.30000000	16.93041	2.36959	12.27764
SWAPTION.USD.Rec.3Y.1Y.LIBOR.3M.T3750.R0.0	Yield	28.60000000	28.98757	-0.38757	-1.35513
SWAPTION.USD.Rec.3Y.5Y.LIBOR.3M.T3750.R0.0	Yield	23.70000000	22.37392	1.32608	5.59527
SWAPTION.USD.Rec.3Y.10Y.LIBOR.3M.T3750.R0.0	Yield	20.70000000	19.28350	1.41650	6.84300
SWAPTION.USD.Rec.3Y.20Y.LIBOR.3M.T3750.R0.0	Yield	17.90000000	16.36715	1.53285	8.56341
SWAPTION.USD.Rec.4Y.1Y.LIBOR.3M.T3750.R0.0	Yield	24.70000000	25.29889	-0.59889	-2.42465
SWAPTION.USD.Rec.4Y.5Y.LIBOR.3M.T3750.R0.0	Yield	21.50000000	20.90217	0.59783	2.78062
SWAPTION.USD.Rec.4Y.10Y.LIBOR.3M.T3750.R0.0	Yield	19.30000000	18.33614	0.96386	4.99410
SWAPTION.USD.Rec.4Y.20Y.LIBOR.3M.T3750.R0.0	Yield	16.90000000	15.91143	0.98857	5.84953
SWAPTION.USD.Rec.5Y.1Y.LIBOR.3M.T3750.R0.0	Yield	22.30000000	22.65909	-0.35909	-1.61028
SWAPTION.USD.Rec.5Y.5Y.LIBOR.3M.T3750.R0.0	Yield	19.90000000	19.53814	0.36186	1.81840
SWAPTION.USD.Rec.5Y.10Y.LIBOR.3M.T3750.R0.0	Yield	18.10000000	17.56035	0.53965	2.98152
SWAPTION.USD.Rec.5Y.20Y.LIBOR.3M.T3750.R0.0	Yield	15.90000000	15.53434	0.36566	2.29972
SWAPTION.USD.Rec.7Y.1Y.LIBOR.3M.T3750.R0.0	Yield	19.70000000	19.29682	0.40318	2.04660
SWAPTION.USD.Rec.7Y.5Y.LIBOR.3M.T3750.R0.0	Yield	17.90000000	18.32965	-0.42965	-2.40028
SWAPTION.USD.Rec.7Y.10Y.LIBOR.3M.T3750.R0.0	Yield	16.80000000	16.65664	0.14336	0.85333
SWAPTION.USD.Rec.7Y.20Y.LIBOR.3M.T3750.R0.0	Yield	14.80000000	14.92902	-0.12902	-0.87177
SWAPTION.USD.Rec.10Y.1Y.LIBOR.3M.T3750.R0.0	Yield	16.70000000	19.50375	-2.80375	-16.78893
SWAPTION.USD.Rec.10Y.5Y.LIBOR.3M.T3750.R0.0	Yield	15.60000000	17.04765	-1.44765	-9.27978
SWAPTION.USD.Rec.10Y.10Y.LIBOR.3M.T3750.R0.0	Yield	15.00000000	15.75050	-0.75050	-5.00331
SWAPTION.USD.Rec.10Y.20Y.LIBOR.3M.T3750.R0.0	Yield	13.40000000	14.28795	-0.88795	-6.62649
SWAPTION.USD.Rec.15Y.1Y.LIBOR.3M.T3750.R0.0	Yield	15.90000000	17.51195	-1.61195	-10.13802



4.2.4 Sample Coupon Structures

Dynamic Floater Coupon Structure

100% * 10Y EUR CMS capped at 10 * (10Y EUR CMS – 2Y EUR CMS), floored at 0

CMS Spread Coupon Structure

10% * (10Y EUR CMS – 2Y EUR CMS), floored at 0