FX and Money Market Analytics

Version 11.1

January 2010 — Second edition

Welcome to the Calypso FX and Money Market Analytics guide which aims to provide an understanding of the analytics that underlie the Calypso pricing capability.

Revision date	Edition	Comments
June 2009	First edition	First edition for version 11.0.
January 2010	Second edition	Second edition for version 11.1.

Contents

Section 1. FX and FX Option Conventions 5
1.1 Currency Pair Definitions 5
1.2 Tenor Arithmetic 5
1.2.1 FX Spot Date 5
1.2.2 FX Forward Tenors 6
1.2.3 FX Option Tenors 6
Section 2. FX Curves 7
2.1 Generation from Yield Curves 7
2.2 FX Points Generator 7
2.3 FX Forward Generator 9
Section 3. Interest Rate Curves using FX 10
3.1 Basis Curve Generation 10
3.2 Curve Zero FX Derived Generation 10
3.2.1 Adding Single-Currency Swaps 12
Section 4. FX Volatility Surfaces 13
4.1 Simple and Derived Surfaces 13
4.2 Delta Lookup Procedure 13
4.3 FXOptionDelta Volatility Surface Generator 13
4.3.1 Generator Parameters 14
4.3.2 Delta Quote Types 16
4.3.3 Conversion to Forward Delta 16
4.3.4 Interpolation and Extrapolation Parameters 19
4.3.5 Outright Variance Interpolation 19
4.3.6 Trading Time Interpolation 19
4.3.7 Smile Extrapolation 20
4.4 Interpolation on Generated Surface Points 21
4.4.1 Interpolator3DLinear 22
4.4.2 Interpolator3DLinearExtended 22
4.4.3 Interpolator3DSpline1D 23
4.4.4 FXVolInterpolator 23
Section 5. FX Valuation 24
5.1 FX Spot and Forwards 24
5.1.1 FX Forward Rate Projection from Interest Rates 26
5.1.2 PricerFXForwardHomeBased 27
5.2 FX Swaps 29
5.3 FX Cash 31
5.4 FX NDF 31
5.5 FX Option Forward 31
Section 6. FX Option Valuation 32

6.1 Overview ... 32
6.2 FX Option Pricers ... 33
Section 7. Option Algorithms ... 39
7.1 Garman-Kohlhagen ... 39
7.2 Binomial Single Asset Tree ... 39
7.3 Binary/Digital Options ... 42
7.3.1 Cash or Nothing Call Option ... 42
7.3.2 Cash or Nothing Put Options ... 45
7.3.3 Asset or Nothing Binary Call Option ... 48
7.3.4 Asset or Nothing Put Options ... 51
7.4 Barrier Options ... 55
7.4.1 Simple Barrier Options ... 55
7.4.2 Standard Barrier Options ... 56
7.4.3 Barrier-At-Expiry Options ... 59
7.4.4 Double Barrier-At-Expiry Options ... 60
7.4.5 Digital Single - Barrier Options ... 61
7.4.6 Digital Double - Barrier Options ... 61
7.4.7 Adjustment for Discrete Barriers ... 61
7.5 Asian Options ... 61
7.6 LookBack Options ... 61
7.6.1 Floating Strike Lookback Options ... 61
7.6.2 Fixed Strike Lookback Options ... 62
Section 8. The Vanna-Volga Method ... 63
8.1 The Cost of Vega Hedging ... 63
8.2 The Adjustment Formula ... 64
8.3 Barrier Option Treatment ... 65
8.4 Volatility Interpolation ... 65
8.5 Comparison with Wystup ... 65
8.6 Calypso Barrier Option Pricer ... 65
8.7 Calculation Choices ... 66
8.8 Volatility Interpolation (Future Enhancement) ... 66
8.9 Example ... 66
Section 9. The Heston Model ... 70
9.1 Product Coverage ... 71
9.2 Pricers ... 71
9.3 Pricing Parameters ... 71
9.4 Model Calibration ... 72
9.4.1 Surface Generation with FXOptionDeltaHeston ... 72
9.4.2 Differential Evolution and its Parameters ... 74
9.4.3 Log Category ... 75
9.4.4 Timing ... 75

9.4.5 Lack of Uniqueness 75
9.5 Configuring a Pricer to Use a Calibration 75
Section 10. Money Market Instruments 77
10.1.1 PricerSimpleMM 77
Section 11. Bond and Money Market Futures 80
11.1 PricerFutureBond 80
11.2 PricerFutureMM 80
11.3 PricerFutureOptionBond and PricerFutureOptionMM 80
11.4 Future Option Volatility Surfaces 81
11.4.1 Future Option Generator 81

Section 1. FX and FX Option Conventions

1.1 Currency Pair Definitions

Primary and Quoting Currencies

A Currency Pair is specified by a *primary currency* and a *quoting currency.* The terminology indicates that FX rates for the pair are given in terms of amount of quoting currency per one unit of primary currency. Then for any amount of primary currency, one converts to quoting currency by multiplication:

$$\text{Quoting Currency Amount} = (\text{FX Rate}) * (\text{Primary Currency Amount})$$

Example: For a EUR/USD pair with EUR defined as the primary currency and USD as the quoting currency, a quote of 1.20 indicates 1.0 EUR can be exchanged for 1.20 USD.

However, it is possible to reverse the definition; in the Calypso Currency Pair window one can set the “Divide” choice, reversing the roles of primary and quoting currency for FX rate quotes. Then primary is converted to quoting currency by dividing by the FX rate, rather than multiplying. However, the same effect can be achieved by defining a Currency Pair with the currencies switched. The “Divide” choice is therefore best avoided except in rare circumstances.

Spot Days

A Currency Pair also requires a specification of the number of days between the quotation date (or trade date) and the Spot Date on which the exchange is to take place at that quote. Typically this is two business days.

The Spot Days can be specified in two ways. It can be established directly for each Currency Pair. Or it can be derived from the default Spot Days assigned to each individual currency, in which case the Spot Days for the pair is defined to be the maximum of the Spot Days of the component currencies.

The date on which the exchange takes place is termed the *Value Date* in the FX market. The Value Date is typically the Spot Date, but can also be one business day after the quotation date (for O/N quotes).

Note: Calypso uses the term “Val date” to denote the day of the market data used in pricing a trade. For FX trades this corresponds to the date the quote is made (price is agreed on) and is not the same as the FX market’s “Value Date”, which instead corresponds to what Calypso terms the trade’s *maturity date* or *settle date*.

Holidays

Each currency has a set of holidays associated with it. The holidays of a Currency Pair is the union of the holidays of the two currencies.

1.2 Tenor Arithmetic

The over-the-counter FX and FX Option markets provide quotes according to tenor -- 1W, 3M, 1Y, etc. The computation of dates from these tenors differs between the two markets. The following market conventions are used by Calypso in producing dates for FX trades and FX curve generation.

1.2.1 FX Spot Date

Given a quotation date (trade date) the corresponding Spot Date is computed from the Spot Days of the Currency Pair.

The simplest way to do this is to add the number of Spot Days, as business days, to the quotation date, using the holidays for the pair. Actual market practice is more involved. To use the simple calculation, the Calypso environment variable USE_UPDATED_SPOT_DATE_CALC_B must be set to False.

The following is the market convention Calypso uses for finding the Spot Date. Essentially it takes into account the NYC (New York) holidays even if USD is not one of the currencies of the pair.

1. For each non-USD currency of the pair, add the number of business days equal to the Spot Days to the quote date, using that currency’s holiday calendar. For USD, add nothing to the quote date.

2. Take the latest date calculated in Step 1.

3. If neither currency of the pair is USD, add the NYC (New York) holidays to the holidays defined for the Currency Pair. Exception: If the Currency Pair has its own “third calendar” defined, add in the holidays for that calendar rather than NYC.

4. If the date found at Step 2 is not a holiday in the combined NYC-Currency Pair holiday collection, then find the next following business day for that collection.

1.2.2 FX Forward Tenors

Given a tenor and a trade date for an over-the-counter FX forward trade, the following procedure is used to derive the forward settlement date.

1. Find the Spot Date as described above.

2. Obtain the holiday calendar for the Currency Pair. If neither currency of the pair is USD, then extend the holiday calendar by adding the NYC holidays; or, if the pair has a “third calendar” defined, add that holiday calendar rather than NYC. Use the combined holiday calendar in the following steps.

3. Find a trial forward date:

- If the tenor is less than or equal to 6 days, add that number of business days to the Spot Date using the holiday calendar of Step 2.
- If the number of tenor days is greater than 6, add that number of calendar days to the Spot Date. (1W tenor is 7 calendar days, 1M counts as 30 calendar days, etc.).

4. If the forward date of Step 3 is not a business day:

- If the number of tenor days is less than 28, find the next business day.
- If the number of tenor days is 28 or greater, find the next business day using the Modified Following method. If the tenor is given in whole months and the Spot Date was the last business day of a month, then subsequently roll the forward date to the last business day of its month.

5. The result is the forward date.

1.2.3 FX Option Tenors

Given a tenor and a trade date (global quotation date) for an over-the-counter option, the following procedure is used to derive expiry and delivery dates.

It the tenor is O/N:

1. The expiry date is the next non-weekend day after the trade date. The expiry is allowed to fall on a holiday.

2. The delivery date is the FX Spot Date related to the expiry date, using the Spot Date method described above.

Note: *T/N or S/N tenors are undefined for FX Options.*

If the tenor is expressed in terms of days or weeks:

1. If the tenor is in weeks, convert to days using 7 days per week. Example: 2W is interpreted as 14 days.

2. Add the tenor number of days to the trade date. This is the expiry date. It does not matter if this is a business day or not, as the expiry date is allowed to fall on a holiday for these tenors.

3. Find the FX Spot Date related to the expiry date, using the Spot Date method described above. This is the delivery date.

If the tenor is expressed in terms of months or years:

1. Add the tenor to the spot date to find the delivery date, using the same method as described above for FX Forwards.

2. To find the expiry date, work backward from the delivery date to find the first business day whose associated Spot Date is equal to the delivery date.

Section 2. FX Curves

2.1 Generation from Yield Curves

An FX Curve can be generated from a pair of yield curves using the arbitrage-free relationship between forward FX rates and the discount rates of the two currencies.

In the FX Curve application, the user specifies this generation method by un-checking the "Generate from instruments" box and by selecting yield curves for the "Base Curve" and the "Quote Curve." Pictorially:

(Base (Primary) Currency Yield Curve) * (Quoting Currency Yield Curve) → FX Curve

The generation equations are as follows. Define:

- $D_Q(T_{SPOT}, T)$: Discount factor of quoting currency yield curve from spot date to forward date
- $D_B(T_{SPOT}, T)$: Discount factor of primary currency yield curve from spot date to forward date
- R_{SPOT}: Spot FX rate (number of units of quoting currency per single unit of primary currency)
- $R_{fwd}(T)$: Forward FX rate at time T

All quantities also have BID and ASK superscripts to identify the quote side. The equations are:

$$R_{fwd}(T)^{ASK} = R_{SPOT}{}^{BID} \frac{D_B(T_{SPOT}, T)^{ASK}}{D_Q(T_{SPOT}, T)^{BID}}, \quad T > T_{SPOT}$$

$$R_{fwd}(T)^{BID} = R_{SPOT}{}^{ASK} \frac{D_B(T_{SPOT}, T)^{BID}}{D_Q(T_{SPOT}, T)^{ASK}}, \quad T > T_{SPOT}$$

For times prior to the spot date,

$$R_{fwd}(T)^{ASK} = R_{SPOT}{}^{BID} \frac{D_Q(T, T_{SPOT})^{BID}}{D_B(T, T_{SPOT})^{ASK}}, \quad T < T_{SPOT}$$

$$R_{fwd}(T)^{BID} = R_{SPOT}{}^{ASK} \frac{D_Q(T, T_{SPOT})^{ASK}}{D_B(T, T_{SPOT})^{BID}}, \quad T > T_{SPOT}$$

Note: Using Bid and Ask spreads as inputs can result in unrealistic spreads on the resulting interest rates. For this reason, many market practitioners prefer to use Mid rates and only add in a spread when making a trade.

For more details on the no-arbitrage equations, please see the section "FX Forward Rate Projection from Interest Rates" in FX Valuation chapter.

2.2 FX Points Generator

The FXPoints Generator creates an FX curve based on FX swap points. It generates curve dates, uses Forward FX points (spreads) as quote inputs, and adjusts points for O/N, T/N market conventions.

The generator is very simple, as the swap points are the forward FX points for dates after the spot date. The only calculations are:

- calculation of the settlement dates for the given underlying tenors
- application of the market conventions for O/N and T/N swap points.

- application of the bid/ask market convention for swap points after the spot date.

The rule for swap points after the spot date is:

- If the bid price is greater than the ask price, the quotes are using an implied minus sign. Supply the sign before placing the values on the curve as forward FX points.

The rules for swap points prior to spot are as follows:

- The sign of the points is reversed.
- Bid and Ask are reversed.
- The all-in rate for exchange today is found by summing the O/N and the T/N quotes, if the spot date is two business days from today. For markets in which the spot date is only one business day (e.g., USD/CAD), the O/N quote alone gives today's points.

Reference: FX quote conventions are detailed in *Mastering Financial Calculations*, R. Steiner (Prentice Hall, 1998).

Example. The following are the Quotes for the FX Swap underlyings and the FX Spot Rate. Because the FX Points generator is used, the quotes are expected to be expressed using the market convention of FX Swap points. (Only mid quotes are used here, so the bid/ask sign conventions will not apply.)

FX.USD.JPY.ON	-0.91
FX.USD.JPY	109.1
FX.USD.JPY.TN	-0.32
FX.USD.JPY.1D	-2.28
FX.USD.JPY.1M	-11.205
FX.USD.JPY.2M	-21.85
FX.USD.JPY.3M	-31.95
FX.USD.JPY.6M	-67.56
FX.USD.JPY.9M	-111.34
FX.USD.JPY.1Y	-164.14

From these FX Swap points, the following FX Curve is created on December 5, 2003, whose spot date is December 9, 2003:

Date	Offset	Mid
12/5/2003	0	1.23
12/8/2003	3	0.32
12/9/2003	4	0
12/10/2003	5	-2.28
1/9/2004	35	-11.205
2/9/2004	66	-21.85
3/9/2004	95	-31.95
6/9/2004	187	-67.56
9/9/2004	279	-111.34
12/9/2004	370	-164.14

Here the resulting points are Forward Points, that is, the simple difference from the FX Spot Rate. To find the all-in forward rate, one uses:

$$\text{All} - \text{in Rate} = (\text{Spot Rate}) + (\text{Points on FX Curve}) / (\text{Points Factor}).$$

So for example, the rate for a USD/JPY exchange that settles on 3/9/2004 is

109.1 + (-67.56) / 100 = 108.4244.

The same applies to dates before spot, that is, there is no FX Swap quote convention in the output. So for an exchange that settles today, December 5, 2003, the rate is

109.1 + 1.23/ 100 = 109.1123.

2.3 FX Forward Generator

The FXForward Generator creates an FX curve based on FX Forwards all-in rates. It generates curve dates and spreads as the difference between FX Forward quotes and Spot quotes.

Since FX Forward all-in rates are not the standard quotation method in the marketplace, this generator is less useful than the FXPoints.

Section 3. Interest Rate Curves using FX

FX rates can be used as input to the generation of interest rate curves. The inputs and the result of two types of curve creation are shown in the following table.

Generation Name	Currency 1 curves	Currency 2 curves	Currency 1 vs. Currency 2	Other Allowed Quotes	Resulting Interest Rate Curve
Curve Zero FX Derived	Interest Curve	None	FX Curve	Single-currency swaps	Interest curve in the form of a Zero Curve
Curve Basis	Interest curve (to project both discount and forward rates)	Interest curve (for discounting only)	FX Forward spreads; Cross-Currency Basis Swaps	Any curve underlying	Interest curve in the format of a Basis Curve (for forward rate generation)

3.1 Basis Curve Generation

The basis curve is the most flexible way of producing an interest curve from cross-currency information. Basis curve generation is described fully in the Calypso document, "Interest Rate Derivatives Analytics."

3.2 Curve Zero FX Derived Generation

Curve Zero FX Derived generates an interest rate curve of the quoting currency from the interest rate curve of the primary currency and an FX curve. Pictorially:

(Primary Currency Yield Curve) * (Primary/Quoting FX Curve) → Quoting Currency Yield Curve

The yield curve is derived using the arbitrage-free relationship between forward FX rates and the discount rates of the two currencies. Using the same quantities as described above in the section on "Generation from Zero Curves", the no-arbitrage relations are:

$$D_Q(T_{SPOT},T)^{BID} = \frac{R_{SPOT}^{BID}}{R_{fwd}(T)^{BID}} D_B(T_{SPOT},T)^{ASK}, \quad T > T_{SPOT}$$

$$D_Q(T_{SPOT},T)^{ASK} = \frac{R_{SPOT}^{ASK}}{R_{fwd}(T)^{ASK}} D_B(T_{SPOT},T)^{BID}, \quad T > T_{SPOT}$$

For the case where $T < T_{SPOT}$, the discount factors are inverted, resulting in:

$$D_Q(T,T_{SPOT})^{BID} = \frac{R_{fwd}(T)^{BID}}{R_{SPOT}^{BID}} D_B(T,T_{SPOT})^{ASK}, \quad T < T_{SPOT}$$

$$D_Q(T,T_{SPOT})^{ASK} = \frac{R_{fwd}(T)^{ASK}}{R_{SPOT}{}^{ASK}} D_B(T,T_{SPOT})^{BID}, \quad T < T_{SPOT}$$

Be aware that using Bid and Ask spreads as inputs can result in unrealistic spreads on the resulting interest rates. For this reason, many market practitioners prefer to use Mid rates and only add in a spread when making a trade.

For more details on the no-arbitrage equations, please see the section “FX Forward Rate Projection from Interest Rates” in the chapter on FX Valuation.

Example. Suppose the following is the zero curve for USD Libor 6M generated on December 5, 2003, with zero rates expressed as Act/360 Annual rates:

Date	Offset	Zero	DF
12/8/2003	3	1.03275	0.999914
12/9/2003	4	1.03275	0.999886
1/9/2004	35	1.15986	0.998879
2/9/2004	66	1.16701	0.997875
3/9/2004	95	1.17879	0.996912
6/9/2004	187	1.27864	0.993422
9/9/2004	279	1.43655	0.989007
12/9/2004	370	1.6233	0.983586
12/9/2005	735	2.38331	0.953049
12/11/2006	1,102	3.01428	0.913102
12/10/2007	1,466	3.48916	0.869651

These discount factors are given as of the curve generation date (Dec 5 2003), but in the FX equations one needs the discount factor based on the FX Spot date, Dec 9 2003. Denoting the curve generation date by T_0, one has:

$$D_B(T_{SPOT},T) = \frac{D_B(T_0,T)}{D_B(T_0,T_{SPOT})}, \quad T > T_{SPOT}$$

The generation also requires a USD/JPY FX curve; for this, use the example curve described above under “FX Points Generator.” The quoting curve calculation is shown in the following table:

Date	FX Fwd Points	FX Forward Rate	Primary Ccy DF	Primary Ccy DF as of Spot	Quoting Ccy DF as of Spot	Quoting Ccy DF
12/5/2003	1.23	109.1123	1	0.999886	0.999998578	1
12/8/2003	0.4	109.104	0.99991438	0.999971	1.00000813	0.99999
12/9/2003	0	109.1	0.99988585	1	1	0.999999
12/10/2003	-2.28	109.0772	*0.99985674*	0.999971	1.000179907	1.000178
1/9/2004	-11.205	108.98795	0.99887948	0.998994	1.000020576	1.000019
2/9/2004	-21.85	108.8815	0.99787513	0.997989	0.999991784	0.99999
3/9/2004	-31.95	108.7805	0.99691227	0.997026	0.999954453	0.999953
6/9/2004	-67.56	108.4244	0.99342201	0.993535	0.9997262106	0.999725
9/9/2004	-111.34	107.9866	0.98900682	0.98912	0.999318085	0.999317
12/9/2004	-164.14	107.4586	0.98358625	0.983699	0.998724259	0.998723

For example, the calculation for 6/9/2004 (June 9 2004) is:

$$D_B(T_{SPOT},T) = \frac{0.99342201}{0.99988585} = 0.993535422.$$
$$R_{SPOT} = 109.1, \quad R_{fwd}(T) = 108.4244$$
$$D_Q(T_{SPOT},T) = \frac{109.1}{108.4244} 0.993535422 = 0.9997262106.$$
$$D_Q(T_0,T) = 0.9997262106 * 0.999999 = 0.999724789.$$

3.2.1 Adding Single-Currency Swaps

Single-currency swaps in the currency of the result curve can be added as underlyings. Using bootstrapping, these swaps will extend the result curve beyond the points generated using the FX Curve.

The bootstrap generator that is used will be the same type as used in the interest curve of the other currency. For example, taking the USD/JPY case, if one has a USD interest rate curve that was generated with the Bootstrap Forwards Generator, then for the FX Derived curve the JPY swaps will extend the resulting JPY curve also using the Bootstrap Forwards algorithm. If the USD had no generator assigned to it, then the default used for the JPY swaps is the standard Bootstrap Generator.

In keeping with the convention in the usual Bootstrap generator, the single-currency swaps take precedence over money market instruments. For example, suppose the FX Curve was created out to 1Y using 1Y FX Forwards, so

that the cash part of the derived curve goes out to 1Y. If one wants to also use swaps starting with a 9M swap, then points of the cash curve after the 9M date are removed, so that all the swap information will be used.

Section 4. FX Volatility Surfaces

4.1 Simple and Derived Surfaces

FX Volatility Surfaces can be input manually by the user, loaded from an external source, or generated from market quotes using a Calypso Volatility Surface generator. The strike axis can be defined in terms of Absolute Strike, Delta, or ATM-Relative Strike.

The Calypso *FXOptionDelta* generator captures the conventions of the over-the-counter market and provides a variety of methods for interpolation on volatilities. The underlying trades of the surface can include Calls/Puts, Risk Reversals, Butterflies and Strangles.

4.2 Delta Lookup Procedure

If the volatility surface has been created using a Delta axis, the volatility for any given option is found by an iterative procedure. Define:

$V(\delta, T)$: the volatility of the surface at delta value D and expiry T

$D(\sigma, T)$: the calculated delta of the option given a volatility σ, expiry T

For a vanilla European option, $D(\sigma, T)$ is the Black-Scholes Forward Delta, assuming the usual case where the volatility surface is calibrated in Forward Delta.

To find the volatility of a vanilla European option, the following equation is solved for σ:

$$V(D(\sigma, T), T) = \sigma$$

When this is satisfied, the delta and volatility of the option are consistent with the volatility surface. The solution of this equation is found using an iterative root-finding numerical method.

4.3 FXOptionDelta Volatility Surface Generator

The **FXOptionDelta Generator** computes Put and Call volatilities from market quotes for at-the-money options (ATM), Risk Reversals, and the Strangles or the Butterflies. The quotes are in the form of volatility for a specified delta.

The delta can be either with or without premium, that is, f options are usually valued by the market using the quoting currency (in points), the standard Black-Scholes delta is used in quoting volatilities, while if the market uses the primary currency (as percent of notional), the Black-Scholes delta minus the premium (option value) is used in quotation. This choice is set by a parameter on the generator for a particular currency pair.

Quotes can also be quoted with respect to either spot delta or forward delta. A generator parameter allows specifying an expiry tenor -- for example, 1Y -- beyond which the quotes are interpreted as forward delta, and before which they are regarded as spot delta.

The output of the generation is always in the form of volatilities for forward delta "without premium." Interpolation is done on these forward deltas. Thus the volatility surface that calls this generator must have strike axis of type "Delta."

This generator also supports of spline extrapolation of delta. A set of parameters specify the extrapolation points in terms of the slope of the known delta smile. A spline is drawn through the extrapolation points before any interpolation is performed. This ensures the same spline is used for interpolation as for extrapolation.

The underlying objects associated with the quotes are VolSurfaceUnderlyingFXOpts. For each expiry, to find the Put volatility, P, and Call volatility, C, the following quoting conventions are assumed:

ATM = Volatility of a zero-delta straddle

Risk Reversal = C – P (if incoming quotes are P – C, they need to be reversed)

Strangle = (P + C) / 2

Butterfly = (P + C) / 2 - ATM.

In the marketplace the direction of the Risk Reversal can be either C – P or P – C, as indicated by an additional symbol, usually C or P, respectively. If quotes are P – C they need to be multiplied by -1 before entering into

Calypso. The ATM is the volatility of a zero-delta straddle, a long call and long put at the same strike where the strike is that at which the delta of the put is equal and opposite to the delta of the call.

The following put-call parity relation is assumed for the forward delta of puts and calls at the same strike and expiry:

Call Delta % = 100 - Put Delta %

For a given maturity, for a forward delta of 50 only the ATM volatility is needed (call and put have same volatility at-the-money). For other values of delta, for each delta and expiry there is required one quote each of Risk Reversal and Butterfly to solve for the Call and Put volatilities, as follows:

Put Volatility = ATM + Butterfly - (1/2)(Risk Reversal)

Call Volatility = ATM + Butterfly + (1/2)(Risk Reversal)

Alternately, a quote of a Risk Reversal and a Strangle is needed:

Put Volatility = Strangle - (1/2)(Risk Reversal)

Call Volatility = Strangle + (1/2)(Risk Reversal).

Example

Suppose one is given the following quotes for USD/JPY options on July 15 2004:

Note the Risk Reversal quotes are in P – C form. If the delta are interpreted as forward delta (see below), the resulting volatility surface for 1M, 6M, and 1Y is:

Tenor	*Exp Date*		*Call Delta*	
		25	**50**	**75**
1M	Aug 19 2004	9.515 / 9.815	9.4 / 9.7	9.815 / 10.115
6M	Jan 19 2005	8.74 / 8.99	9.0 / 9.25	9.89 / 10.14
1Y	Jul 15 2005	8.665 / 8.815	9.0 / 9.15	10.015 / 10.165

4.3.1 Generator Parameters

The FXOptionDelta generator has a number of parameters to define the interpretation of the input quotes and tenors and the interpolation/extrapolation of the output volatilities. These are summarized below, with more details given in the subsequent sections.

DATE_CUT

Identifies the DateCut (partial day) in which the expiry time of the underlying options fall. Used for determining the weighted trading times between the generation datetime and the expiry datetimes.

SPOT_DELTA_END_TENOR

The last Tenor for which the delta in the quotes are interpreted as spot delta. All expiry dates on or before this tenor are interpreted as spot delta; strictly after this tenor are forward deltas. Can be zero (so that all deltas are forward) or extremely large (so that all deltas are spot).

IS_DELTA_WITH_PREMIUM

Identifies whether the deltas in the quotes are interpreted as delta with premium or as delta plain (ordinary Black-Scholes delta).

INTERPOLATE_OUTRIGHT_VARIANCE

Identifies whether interpolation will be done on the outright variance (the square of the volatility times the calendar time) rather than on the volatility.

INTERPOLATE_TRADING_TIME

Identifies whether interpolation over time is done in weighted trading time rather than simple calendar time.

ROLL_CALENDAR_VOL

The relationship between Trading Time Volatility and Calendar Time Volatility is

(calendar day volatility)2 * calendar days = (trading time volatility)2 * (trading time)

If ROLL_CALENDAR_VOL is set to true and trading time shifts, then trading time volatility is held constant and the calendar day volatility is adjusted according to the above relationship. Therefore, if INTERPOLATE_TRADING_TIME is set to false, the setting for ROLL_CALENDAR_VOL is inapplicable and should be set to false.

FX_DATE

FX_Date is the current calendar date used by the FX Market to determine tenors such as Spot, 1 Month, etc. The FX Date normally advances to the next business day at 5 PM New York time, but the volatility surface window allows the trader to override this date.

GRANULARITY

The granularity is to define the trading time associated to each tenor. The choices are either Daily or Continuous. If daily is chosen and weighting is set to false, then trading time is equal to calendar time.

UP_EXTRAP_1_DELTA

When the smile is turned upward at one or another end, this extrapolation parameter multiplies the slope of the first two points on the surface to find the volatility of the 1.0 call delta, and/or multiplies the slope of the last two points to find the volatility of the 99.0 call delta. Usually this is a positive number to continue the smile direction upward.

DOWN_EXTRAP_1_DELTA

When the smile is turned downward at one or another end, this extrapolation parameter multiplies the slope of the first two points on the surface to find the volatility of the 1.0 call delta, and multiplies the slope of the last two points to find the volatility of the 99.0 call delta. Usually this parameter is zero or small to prevent volatilities from going negative.

Default parameter values:

IS_DELTA_WITH_PREMIUM, "false"

SPOT_DELTA_END_TENOR, "1Y"

INTERPOLATE_OUTRIGHT_VARIANCE, "true"

INTERPOLATE_TRADING_TIME,"true" (use "false" for faster interpolation)

DATE_CUT, ""

UP_EXTRAP_1_DELTA, "3.0"

DOWN_EXTRAP_1_DELTA, "0.0"

4.3.2 Delta Quote Types

The market quotes volatility relative to a given delta for each strategy: risk reversal, butterfly, and strangles. There are four types of delta that are used in these quotes:

- **Spot Delta** : The change in spot option premium with respect to the spot FX rate, when the premium is settled in the quoting currency of the FX pair.
- **Forward Delta** : The change in the forward value at expiry of the option premium with respect to the forward FX rate, when the premium is settled in the quoting currency of the FX pair.
- **Spot Delta with Premium :** The spot delta minus the premium per unit notional. It is equal to the change in spot primary-currency premium with respect to the spot FX rate, followed by multiplication by the spot FX rate.
- **Forward Delta with Premium :** The forward delta minus the forward value of the premium per unit notional. It is equal to the change in the forward value at expiry of the primary-currency premium with respect to the forward FX rate, followed by multiplication by the forward FX rate.

For Spot Delta and Forward Delta, where premium is in quoting currency, the market often expresses the price in terms of quoting currency points per each primary currency point, and so is called the *Points* quotation convention. For Spot Delta with Premium and Forward Delta with premium, where the premium is in primary currency, the market can express the premium as a percentage of the primary currency notional value, and so is called the *Percentage* quoting convention.

The type of the delta quote is specified with the two generator parameters SPOT_DELTA_END_TENOR and IS_DELTA_WITH_PREMIUM. The next section gives examples of their use.

4.3.3 Conversion to Forward Delta

For the purposes of rapid volatility lookup and consistent interpolation across the time and strike axes, each delta type is converted to forward delta (without premium) by the FXOptionDelta generator. Then lookup of volatility by the pricer is always done in forward delta space, as described in the section describing the Delta Lookup Procedure.

To demonstrate the conversion to forward delta, consider the following market data:

Tenor	6 months
ATM	10.00
RR 25 Delta	2.00
BFly 25 Delta	5.00

The delta used in this table could be any one of the four types, which would be determined for the volatility surface by the parameter settings. The following illustrates the conversion of each type to forward delta.

Spot Delta Quotes

The quotes are against Spot Delta (without premium) if the parameters have the following settings:

SPOT_DELTA_END_TENOR = 1Y

IS_DELTA_WITH_PREMIUM = False

The quote tenor is 6 months, which is less than the parameter cutoff tenor for the spot delta of 1Y; so the quote is interpreted as spot.

Using the quote conventions for risk reversals and butterflies, the market data corresponds to the following volatilities:

Delta	*Call Volatility*	*Put Volatility*
ATM	10	10
25	16	14

Now use the relationship

$$ForwardDelta = \frac{SpotDelta}{DiscountFactor(Pmry)}$$

where the discount factor uses the primary currency discount rate from the spot date to the expiry date. Suppose this discount factor were 0.98. Then, using the fact that the ATM spot delta will convert to the ATM forward delta of 0.5,

Forward Delta	*Call Volatility*	*Put Volatility*
50	10	10
25.5102	16	14

In practice the volatility surface is only expressed in terms of Call Volatility, which is done using the Put-Call Parity relationship of forward deltas, Call Delta = 100 - Put Delta. So the output for interpolation purposes is:

Forward Delta	*Call Volatility*
25.5102	16
50	10
74.4898	14

The strike axis is now in a form so that looking up volatility by forward delta is now simply a matter of interpolation.

Forward Delta Quotes

If the parameters have the following settings:

SPOT_DELTA_END_TENOR = 0D

IS_DELTA_WITH_PREMIUM = False

then the quotes are against Forward Delta (without premium), because the tenor is greater than the spot tenor cutoff of zero days. Then no further conversion is needed, and one has the output for interpolation:

Forward Delta	*Call Volatility*
25	16
50	10

75	14

Spot-Delta-with-Premium Quotes

For the settings SPOT_DELTA_END_TENOR = 1Y and IS_DELTA_WITH_PREMIUM = True, the 6 month quotes are interpreted as Spot Delta with Premium. To convert to Forward Delta without premium one must first solve for the strike and the corresponding premium. For example, suppose one had:

Currency Pair : USD/JPY

Spot FX Rate : 110.00

JPY Rate : 1%

USD Rate: 4%

Then by iterative search one can find that a 6 month Call option on USD with volatility of 16% will have a Spot-Delta-with-Premium of 25% if the option has a strike of about 116.83. That is, an option with that strike has a Spot Delta (without premium) of 26.76% and a premium of 1.76% of primary notional amount. Then using the USD discount factor of about 0.98 one finds the Forward Delta (without premium) of 26.76/0.98 = 27.3061.

Similarly, the put option with 14% volatility solves to a 101.47 strike, giving Spot Delta without premium of -23.54%, premium of 1.45%, for a Spot-Delta-with-Premium of -25%. The corresponding Forward Delta without premium is 24.0204. The Forward Delta of the call at the same strike and volatility is then 75.9798.

For the ATM, one finds by iteration that at a strike of about 108.07 the put and call have the same magnitude of Spot-Delta-with-Premium -- about 48.87. For the call, one has a premium of 2.92% of primary notional, giving a Spot Delta (without premium) of 51.79, and so a Forward Delta of 52.85. For the put, the premium is 2.66%, giving a Spot Delta of -46.21 and a Forward Delta of -47.15, also corresponding to a call Forward Delta of 52.85.

The result is the following output table for interpolation:

Forward Delta	***Call Volatility***
27.3061	16
52.85	10
75.9798	14

Forward-Delta-with-Premium Quotes

For the settings SPOT_DELTA_END_TENOR = 0D and IS_DELTA_WITH_PREMIUM = True, the 6 month quotes are interpreted as Forward-Delta-with-Premium. The conversion to Forward Delta without premium requires solving for the strike rate and otherwise proceeds as with the preceding case.

At a volatility of 16%, a strike rate of 117.05 for a call gives the desired Forward-Delta-with-Premium of 25%. Computing the Forward Delta without premium for this strike and volatility gives 26.78. The same procedure for a put with volatility 14% produces a strike of 101.31 and a Forward Delta without premium of 23.53, corresponding to 76.47 for a call.

At the ATM volatility of 10% the Forward-Delta-with-Premium has the same magnitude (49.87) for the put and the call at a strike of about 108.062. Computing the Forward Delta without premium for this strike and volatility gives 52.894 for a call, 47.106 for a put. The result is:

Forward Delta	***Call Volatility***
26.78	16
52.894	10
76.47	14

4.3.4 Interpolation and Extrapolation Parameters

A surface that that uses the FXOptionDelta generator will perform interpolation over Forward Delta space. Given an option trade with unknown delta, the iterative procedure is performed as described in the section “Delta Lookup Procedure.”

The general approach to two-dimensional lookup is that described in the section on surface interpolation. However, the FXOptionDelta generator adds new behavior using several parameters:

INTERPOLATE_OUTRIGHT_VARIANCE

INTERPOLATE_TRADING_TIME

UP_EXTRAP_1_DELTA

DOWN_EXTRAP_1_DELTA

These are described in the following sections.

4.3.5 Outright Variance Interpolation

If INTERPOLATE_OUTRIGHT_VARIANCE is set to True, then prior to interpolation each volatility *v* on the surface is transformed into an outright variance, using the definition:

$$\text{OutrightVariance} = v^2 t,$$
$$t = \frac{\text{Days from generation date to expiry date}}{365}$$

Linear or spline interpolation with respect to delta and time is performed on the outright variances. The resulting outright variance is converted back into a volatility, which is then the result of the interpolation.

If INTERPOLATE_OUTRIGHT_VARIANCE is set to False, interpolation is done over the volatilities in the surface as usual.

4.3.6 Trading Time Interpolation

If INTERPOLATE_TRADING_TIME is set to True, the time axis of the volatility surface is converted into *trading time*. Trading time is a weighted time that takes into account traders’ estimates of the speed at which the market will move on various calendar days. For example, weekends and holidays can be given very little weight to reflect the absence of trading. When looking up volatility for a given expiry date of an option, the date is translated into number of trading days forward, which is then used to interpolate on the time axis.

If INTERPOLATE_TRADING_TIME is set to False, no transformation in time is done. The volatility is simply interpolated at the given expiry date using calendar time.

Calculating Number of Trading Days

To calculate trading days one first defines a set of *Cuts* that slices the 24-hour day into two or more segments. One Cut will usually cover the most active trading in the 24-hour day for a given currency pair, while the other Cut(s) will be relatively inactive. A *DateCut* is a combination of a calendar date and a choice of Cut. Each DateCut for an arbitrary period into the future can be given a weight to reflect the expected activity. DateCuts that fall on holidays or weekends can be given little weight, while DateCuts that include scheduled market moving events, such as government reports, can be given higher trading weights. (These weights are stored in weighting tables without regard to a particular volatility surface.)

Then the number of trading days between two DateCuts dc1 and dc2 (where dc1 < dc2) is defined as

$$\mathrm{w} = \sum_{dc1 < dc \le dc2} \mathrm{w}_{\mathrm{dc}} \text{ ,}$$

where w_{dc} is the weight for the DateCut dc.

Trading Day Volatility and Calendar Day Volatility

The volatility that is derived directly from the market quotes is called the *Calendar Day Volatility*. Given a definition of trading time, one can calculate an associated *Trading Day Volatility* by requiring that the outright variance of the two time conventions be the same:

$$\sigma_{trading\_days} \times \sqrt{trading\_days} = \sigma_{calendar\_days} \times \sqrt{calendar\_days}$$

Regardless of the time interpolation, the volatility that results from a surface interpolation is always given in the form of Calendar Day Volatility.

Time Extrapolation

Extrapolation on the time axis is constant after the last tenor in the surface.

To look up volatilities for times prior to the first tenor in the surface, constant extrapolation is again used if either INTERPOLATE_OUTRIGHT_VARIANCE or INTERPOLATE_TRADING_TIME is set to False.

If both INTERPOLATE_OUTRIGHT_VARIANCE and INTERPOLATE_TRADING_TIME are set to True, extrapolation before the first tenor is calculated by requiring the slope of the outright variance with respect to trading time is constant in the extrapolation region, and zero at the surface generation date. Let T_1 be the calendar days to the first tenor, t_1 the trading days, and V_1 the Calendar Day Volatility. Then for calendar days $T < T_1$, corresponding to trading days $t < t_1$, the volatility is extrapolated to be:

$$V(T) = V_1 \sqrt{\frac{(T_1 / t_1)}{(T / t)}}$$

For example, if the first tenor is 1W, weekends are given zero weights and there are no other holidays or special events, there are 7 calendar days and 5 trading days. Then the O/N volatility equals (1W vol) *sqrt(7/5).

4.3.7 Smile Extrapolation

For a given tenor on the surface, denote the known volatility points by $V(d_i)$ for i = 1,2,...N, where di are the Forward Deltas for Calls in increasing order. For example, one might have $d_1 = 10$, $d_2 = 25$, $d_3 = 50$, $d_4 = 75$, and $d_6 = 90$. Extrapolation on the delta axis for a delta $d < d_1$ or $d > d_6$ is accomplished by adding new points to the axis and extending the interpolation method – spline or linear – to pass through these additional points.

The first points are added at the 1% Forward Deltas for Calls and Puts. In terms of Calls alone this corresponds to the 1% and 99% Forward Deltas.

The volatilities at the added points are determined by the generator parameters UP_EXTRAP_1_DELTA and DOWN_EXTRAP_1_DELTA. These specify the slope of the extrapolation from the known volatilities. If the volatility smile at the last two known points is up (volatility versus delta is concave upward), then UP_EXTRAP_1_DELTA is used, and DOWN_EXTRAP_1_DELTA otherwise. Suppose the value of the appropriate parameter is *p*. Then the volatility at the 1% Delta is defined to be

$$V(1\%) = V(d_1) + p * (V(d_1) - V(d_2)).$$

At the other end of the smile, the volatility at 99% Call Delta is defined to be

$$V(99\%) = V(d_N) + p * (V(d_N) - V(d_{N-1})).$$

If spline interpolation is used for the intermediary points one can get an unrealistically fluctuating surface if the extrapolation is too sharp. To guard against this, additional points are added to smooth the spline. These smoothing points are added at deltas half way between the 1% delta and d_1, and half way between d_N and 99%. Define these as:

$$d_A = 0.5 * (1 + d_1)$$
$$d_b = 0.5 * (d_N + 99)$$

The volatility at these deltas is defined by using the average of the slopes of the original portion of the smile and the portion extrapolated in the previous step.

$$V(d_A) = V(d_1) - m*(d_1 - d_A),$$
$$m = 0.5*(\frac{V(d_1) - V(1\%)}{d_1 - 1\%} + \frac{V(d_2) - V(d_1)}{d_2 - d_1}),$$

$$V(d_B) = V(d_N) - n*(d_N - d_B),$$
$$n = 0.5*(\frac{V(d_N) - V(99\%)}{d_N - 99\%} + \frac{V(d_{N-1}) - V(d_N)}{d_{N-1} - d_N}).$$

Then the spline (or line) for interpolation is drawn through the points (1%, V(1%)), (d_A, V(d_A)), (d_1, V(d_1))… (d_N, V(d_N)), (d_B, V(d_B)), (99%, V(99%)). For delta less than 1% or greater than 99%, constant extrapolation is used.

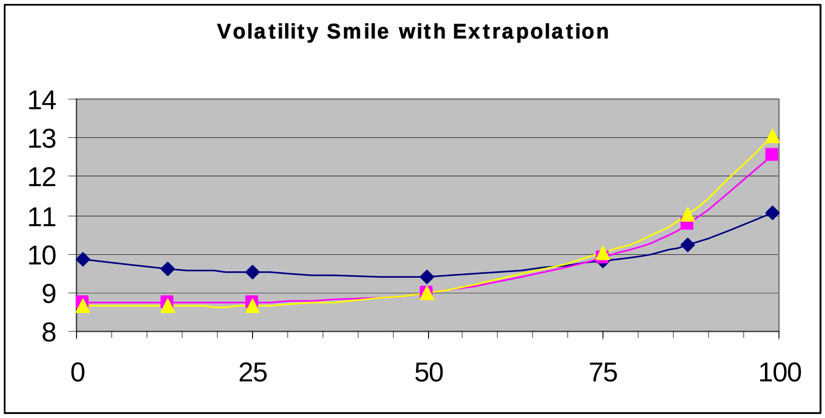
Example. Using the example quotes and volatility surface at the beginning of this chapter, suppose the parameters are given as:

UP_EXTRAP_1_DELTA = 3.0

DOWN_EXTRAP_1_DELTA = 0.0

Then the extrapolation below 25 delta and above 75 delta gives the (bid price) surface for interpolation:

Tenor	*Exp Date*				*Call Delta*			
		1	***13***	***25***	***50***	***75***	***87***	***99***
1M	Aug 19 2004	9.86	9.62885	9.515	9.4	9.815	10.22585	11.06
6M	Jan 19 2005	8.74	8.74	8.74	9	9.89	10.7711	12.56
1Y	Jul 15 2005	8.665	8.665	8.665	9	10.015	11.01985	13.06



Note that the DOWN_EXTRAP_1_DELTA parameter is used on the left end for 6M and 1Y, because the volatility at 25 delta is less than that of the 50 delta – the smile is turned down through those points. Since the value of this parameter is 0, the result is constant extrapolation below 25 for these tenors. In the other cases, the smile is up through the last known points, and the UP_EXTRAP_1_DELTA parameter is used to determine the end points.

4.4 Interpolation on Generated Surface Points

When a surface has had its points generated in terms of forward call deltas, the FX volatility surface interpolators perform straightforward interpolation on these points over two dimensions: the strike axis (usually delta) and the time to expiry.

To interpolate volatility for a vanilla European option with time to expiry T and strike K:

1. Locate on the surface tenors that bracket the time to expiry of the option. If these times are called T1 and T2, one has T1 <= T <= T2. In the simplest case, T = T1, and T2 can be ignored.

2. For time T1, look up the volatility at the option's Forward Delta, D, using the iterative Delta Lookup Procedure. Interpolation on the delta axis is used in this step; this may be either linear or spline interpolation. This produces a volatility V(D, T1). If T = T1, this is the result and nothing further needs to be done; otherwise proceed to the next step.

3. Repeat the procedure at T2 to find another volatility, V(D, T2).

4. Interpolate linearly on the time axis between V(D,T1) and V(D,T2) to arrive at V(D,T).

Extrapolation is used as needed, if D or T does not fall between the first and last points on the respective axes. Constant extrapolation is used as a default. Other extrapolation can be done by specific interpolators or volatility generators.

The following interpolators are available. (Note on terminology: While all the interpolators have "3D" in their name, for FX option volatility uses only two of these dimensions. The third – tenor -- is not applicable to the FX spot underlying of the option.)

4.4.1 Interpolator3DLinear

Interpolator3DLinearExtended provides linear interpolation within the upper and lower boundaries of the calculated time and strike axes. Beyond those boundaries, constant extrapolation is used, unless overridden by the generator's extrapolation parameters.

4.4.2 Interpolator3DLinearExtended

Interpolator3DLinearExtended provides linear interpolation within the bounds of the specified time (expiry) and delta axes; beyond those bounds, the linear extrapolation is used in the delta direction until specified delta values are reached, beyond which constant extrapolation is used. Beyond the bounds of the time axis, constant extrapolation only is used.

The delta axis is defined to have a constantLowerLimit and a constantUpperLimit. Suppose the known axis points are x1, x2,…, xn. If the constantLowerLimit > x1 it is ignored, and a constantUpperLimit < xn is ignored. So consider the case of constantLowerLimit < x1 and constantUpperLimit > xn.

Values of the axis that lie below the constantLowerLimit use constant extrapolation, that is, the result is equal to the result at the constantLowerLimit point. Values of the axis that lie on or between the constantLowerLimit and x1 are computed by linear extrapolation from the first two known points, x1 and x2. Values of the axis that lie between x1 and xn use linear interpolation, from xn to constantUpperLimit use linear extrapolation again, and then values above the constantUpperLimit use constant extrapolation from that point.

The constantLowerLimit and constantUpperLimit are negative or positive infinity by default, meaning there is only linear extrapolation. The user must set the limits in order to cut off linear extrapolation and begin constant extrapolation.

Example: Suppose the volatility surface for a certain expiry is holds these volatilities and deltas:

Call Delta	Volatility
10	20
25	18
50	15
75	19
90	21

Suppose further the surface had specified:

- Constant Lower Limit = 5
- Constant Upper Limit = 95

The following interpolations will result for selected deltas:

Call Delta	Interpolated Vol

2	20.93
5	20.93
15	19.33
95	21.67
97	21.67

4.4.3 Interpolator3DSpline1D

Interpolator3DSpline1D provides spline interpolation along one dimension and linear interpolation in the other dimensions. For FX Options, the spline is used for the delta axis and linear interpolation for the expiry time axis. Constant extrapolation only is used beyond the known points in both dimensions.

The procedure is illustrated below. The "strike" axis for FX options is the delta axis. Define

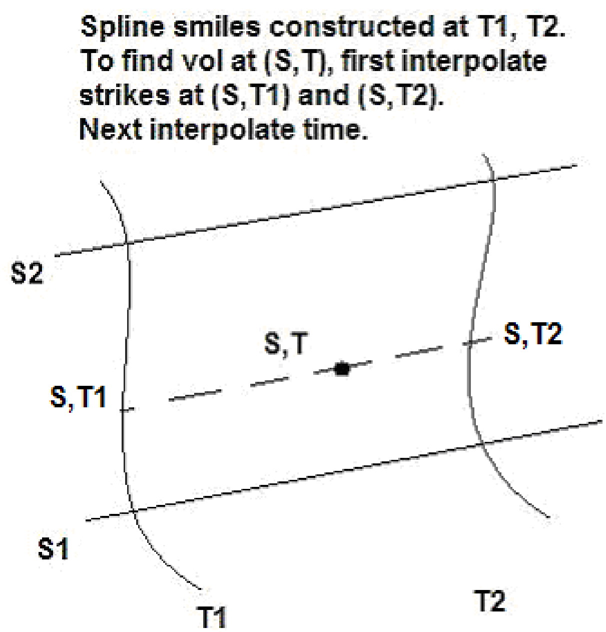
V(S,T) : Volatility at delta strike S and expiry time T

For each expiry time in the surface, a spline is constructed along the delta axis. To find the volatility at any delta and expiry (S,T), first points on the surface are found so that

S1 <= S < S2, T1 <= T < T2.

4.4.4 FXVolInterpolator

FXVolInterpolator provides spline interpolation along one dimension and linear interpolation in the other dimensions. For FX Options, the spline is used for the delta axis and linear interpolation for the expiry time axis. Constant extrapolation only is used beyond the known points in both dimensions. Prior to interpolation each volatility *v* on the surface is transformed into an outright variance.



The delta strike axis is interpolated on first. Using the splines created at expiry times T1 and T2, one finds the volatilities V(S, T1) and V(S, T2).

Then linear interpolation in the time direction is performed on V(S,T1) and V(S,T2) to obtain V(S,T).

Section 5. FX Valuation

Product	Calypso Product	Pricer
FX Spot	FX, FXTakeup	PricerFX
	FXCash	PricerFXCash
FX Forward	FXForward	PricerFXForward PricerFXForwardHomeBased
	FXNDF	PricerFXNDF
	FXOptionForward	PricerFXOptionForward
	FXOptionSwap	PricerFXOptionSwap
FX Swap	FXSwap	PricerFXSwap

5.1 FX Spot and Forwards

Two pricers are available for pricing FX forward trades, **PricerFXForward** and **PricerFXForwardHomeBased**. For FX spot trades one can use **PricerFX** or **PricerFXForwardHomeBased**. There is presently little difference between PricerFX and PricerFXForward.

Net Present Value

The NPV is the sum of the discounted future cashflows combined into a common currency. There are two possibilities depending on the order in which discounting and currency conversion is done:

A. Discount the cashflows using the zero curves of the two currencies and then convert the results to one currency using the FX rate on the value date.

B. Convert the cashflows to a common currency using forward FX rates on the cashflow payment dates and afterwards perform the discounting using the curve of the resulting currency.

In **PricerFXForward / PricerFX** the FX_POINTS parameter distinguishes these two methods. If FX_POINTS is True, then Method B is used, as the forward points are required for doing the future conversion. In PricerFXForwardHomeBased Method B is used, but if forward FX rates are not available in the market data set then Method A is used.

PricerFXForwardHomeBased discounts all cashflows using the discount curve of the Pricing Environment base currency, referred to in this context as the "home" currency. The home currency can be different from either currencies of the FX trade, which are referred to as the trade "base currency" and the trade "quote currency."

The following table summarizes the NPV methods.

Pricer	NPV method	Market Data Needed
PricerFXForward / PricerFX	IF FX_POINTS is False, discount with each ccy curve before conversion (Method A).	- Base ccy zero curve - Quote ccy zero curve - Spot rate
PricerFXForward / PricerFX	IF FX_POINTS is True, convert to trade quote ccy using forward points before discounting with quote ccy curve (Method B).	- Quote ccy zero curve - FX curve Base/Quote - Spot rate

Pricer	NPV method	Market Data Needed
PricerFXForwardHomeBased	Convert cashflows to the home currency prior to discounting using the home discount curve. If direct forward rates are not available use USD as a middle currency. If insufficient FX data is available, revert to Method A.	- Home ccy (pricing environment base ccy) zero curve - FX curves for converting trade base and quote currencies to Home ccy - FX curve for USD to Home ccy conversion, as needed - Spot rates between Home and trade base and quote ccy's.

Pricing Parameters

There are some pricing parameters of particular interest to the FX pricers.

FX_POINTS

Values: True/False

Implmented by: PricerFXForward, PricerFX.

Not implemented by PricerFXForwardHomeBased, because this pricer by default uses forward FX points.

This parameter is explained in the preceding section on NPV.

- If True: FX forward points from an FX curve are used to convert future cashflows to a common currency prior to discounting.
- If False: Future FX cashflows are discounting using separate zero curves for each currency, and the result converted to a single currency.

ADJUST_FX_RATE

Values: True/False

Implemented by: All currency conversion routines in the system, including trade windows and reports. Transient parameter override is allowed by PricerFXForward and PricerFX.

Transient parameter override is not allowed by PricerFXForwardHomeBased, which by design uses ADJUST_FX_RATE equal to True to compute an NPV.

When currency conversion is done for the present values of cashflows, there are two choices for the FX rate. The first choice is to employ the spot FX rate quoted on the valuation date (the date for which the present value was computed). The second choice is to use "Today's Rate", which is the FX rate that could be applied on the valuation date itself if the currency exchange were to be completed on that day.

The ADJUST_FX_RATE parameter determines whether to adjust the current quoted spot rate to obtain the rate for settlement on the quote date rather than on the spot date (two business days after the quote date).

- If True: "Today's Rate" is used for conversion of currency. This rate is obtained from the quoted spot rate by discounting using the two discount curves for the relevant currencies. The usual no-arbitrage condition is used to obtain the adjusted rate, as discussed in the Appendix. (Note that an alternative method would be to look up Today's Rate from an FX curve, but this choice is not currently available).
- If False: Use the quoted spot rate to convert currency.

Note that the term “valuation date" used here, the date on which NPV is taken, should not be confused with the FX market terminology where “Value Date” is another term for “Settle Date” or “Maturity Date.”

ZD_PRICING

Values: True/False

Implmented by: PricerFXForward, PricerFX.

The choice is not allowed by PricerFXForwardHomeBased, which by design uses the equivalent of ZD_PRICING equal to True.

This parameter also appears in some other Pricers.

- If True: NPV is computed as of the valuation date of the trade window or report. (This is sometimes referred to as the "zero day", hence the term ZD_PRICING.)
- If False: NPV is computed as of the spot settlement date relative to the valuation date.

USE_FX_MID

Values: True/False

Implmented by: All FX pricers.

- If True: The Mid rate of the FX quote is used for all trades.
- If False: The Bid or Ask rate of the FX quote is used depending on the direction of the trade from the point of view of the user. If the base currency is being purchased, the Ask rate is used; if the base currency is being sold, the Bid rate is used. In making this determination the FX rate from the quote set is first converted, if necessary, to the standard from expressing units of quote currency that are equivalent to one unit of base currency.

5.1.1 FX Forward Rate Projection from Interest Rates

Forward rate and “today’s” rate can be projected from interest rates as needed. The forward calculation may be necessary when the FX Curve is missing or needs to be extrapolated.

Define the following:

- R_{spot} – The spot FX rate defined for settling FX deals on spot date T_{spot}.
- R_{fwd} – The forward FX rate on the indicated date.
- D_Q (T1, T2) – The discount factor for discounting quoting currency from any date T2 to earlier date T1.
- D_B (T1, T2) – The discount factor for discounting base currency from any date T2 to earlier date T1.

The forward rate for conversions taking place on a date T (“for Value T”) is projected from two interest rate curves as follows:

- If $T >= T_{spot}$ (forward projection),
  $R_{fwd} = R_{spot} * D_B (T_{spot}, T) / D_Q (T_{spot}, T)$.
- If $T < T_{spot}$ (“backward” projection),
  $R_{fwd} = R_{spot} * D_Q (T, T_{spot}) / D_B (T, T_{spot})$.

In these formulas the MID price of the FX and interest rates is used throughout. Instead, one may wish to take the correct side into account for each quote. This does not necessarily produce a better result, as it can increase the spread beyond what appears in the market.

To take the side into account, note each step in the covered arbitrage procedure:

- Borrow 1 unit of Base currency at the ASK Base interest rate.
- Sell the borrowed unit of Base for Quoting currency at the BID spot FX rate.
- Invest the Quoting currency at the BID Quoting interest rate.

At the maturity of the loans, one will repay the Base currency amount:

$$\frac{1}{D_B(T_{SPOT}, T)^{ASK}}$$

and receive the Quoting currency amount:

$$\frac{R_{SPOT}{}^{BID}}{D_Q(T_{SPOT}, T)^{BID}} .$$

These cashflows are the same as the cashflows of a forward FX transaction to sell Base, buy Quoting, and so can be replicated by selling Base to an FX market maker at the FX forward BID price for buying Base. So by the law of one price,

$$R_{fwd}^{\ BID} = R_{SPOT}^{\ \ \ BID} \frac{D_B(T_{SPOT},T)^{ASK}}{D_Q(T_{SPOT},T)^{BID}}, \quad T > T_{SPOT}$$

Similarly, for the forward ASK price,

$$R_{fwd}^{\ ASK} = R_{SPOT}^{\ \ \ ASK} \frac{D_B(T_{SPOT},T)^{BID}}{D_Q(T_{SPOT},T)^{ASK}}, \quad T > T_{SPOT}$$

To derive a discount factor from a time T back to “today’s” date, T_C, two business days before the Spot Date, both projection formulas are needed. One has for the two time intervals (using mid rates):

$$\mathrm{D_B}\ (\mathrm{T_{SPOTt}}, \mathrm{T}) = \mathrm{D_Q}\ (\mathrm{T_{SPOT}}, \mathrm{T}) \frac{\mathrm{R(T)}}{\mathrm{R_{SPOT}}}$$

$$\mathrm{D_B}\ (\mathrm{T_C}, \mathrm{T_{SPOT}}) = \frac{\mathrm{R_{SPOT}}}{\mathrm{R(T_C)}} \mathrm{D_Q}\ (\mathrm{T_C}, \mathrm{T_{SPOT}}) \ .$$

Multiplying these together and using D_B (T_C, T_{SPOT}) * D_B (T_{SPOT}, T) = D_B (T_C, T) , one has for the total time interval from today to the forward date,

$$\mathrm{D_B}\ (\mathrm{T_C}, \mathrm{T}) = \mathrm{D_Q}\ (\mathrm{T_C}, \mathrm{T}) \frac{\mathrm{R(T)}}{\mathrm{R(T_C)}}.$$

5.1.2 PricerFXForwardHomeBased

This section describes in greater detail the algorithm used by PricerFXForwardHomeBased to compute Net Present Value.

This pricer is designed to serve the following requirements:

- the preference for using the discount curve of the home currency over other yield curves, whether or not the home currency is part of the trade
- the need to find forward rates for currency pairs that are not frequently quoted.

There are potentially four currencies involved in the pricer's computation of NPV:

- Trade currencies, the currencies bought and sold, consisting of the Primary (Base) currency and the Quoting currency
- Home (reporting) currency, used across portfolios for risk management and accounting; this is the Pricing Environment “Base” Currency
- Split currency, either USD or EUR, used to define FX rates for a currency pair that is not directly quoted in the marketplace.

Step 1. Forward cashflow conversion to Home currency.

Each cashflow is converted to the Home currency (whether or not the Home currency is one of the currencies in the trade) at the forward FX rate on the cashflow payment date. The rate is drawn from the FX Curve for the cashflow currency vs. home currency pair:

Rfwd = Rspot + forward points for cashflow payment date

If the FX Curve is missing:

If there is no FX curve for the cashflow currency vs. Home currency, the forward rate is obtained through the FX curves with respect to USD. Thus, to convert from CCY1 to HOME when there is no CCY1/HOME curve, the pricer first converts to USD using the CCY1/USD curve and then to HOME using the USD/HOME curve.

If the CCY1/USD curve is missing, then no conversion is done. The pricer defaults to using the zero curve for CCY1 for discounting.

If the FX Curve is too short:

If the FX Curve does not extend to the date of the cashflow payment, the forward rate is not used. Instead discounting is done with the yield curve for the cashflow's currency.

Note this is equivalent to the "FX_POINTS equals false" method in PricerFXForward. The alternative possibility, extrapolation of the FX Curve using interest rate curves, is not currently implemented.

Step 2. Discount Factor Calculation

The Home currency discount curve is used for all discounting. If the conversion cannot be performed then the cashflow's curve is used for discounting. This is equivalent to falling back on the "FX_POINTS equals False" method in PricerFXForward.

Step 3. Conversion to Trade Quote Currency

The pricer returns the NPV in the trade's quote currency. To convert the present value from the Home currency to trade quote currency, the rate on the valuation date, "Today's Rate", is used. This is obtained when possible from the FX curve relating the Home currency and the quote currency, going through USD if necessary.

Equivalence to ZeroCurveFXDerived

It should be noted that finding NPV with PricerFXForwardHomeBased is nearly equivalent to making use of zero curves that have been derived from the Home currency curve using a CurveFX. A curve created in this way is termed a ZeroCurveFXDerived, and there is a curve generator application of this name. If one creates a set of such zero curves, then using PricerFXForward with FX_POINTS set to False, and ADJUST_FX_RATE and ZD_PRICING set to True, will produce NPV that is almost equivalent to using PricerFXForwardHomeBased.

Differences between the methods because of differing methods for adjusting the spot rate to Today's Rate; and handling of the cases where derived zero curves cannot be created is up to the user.

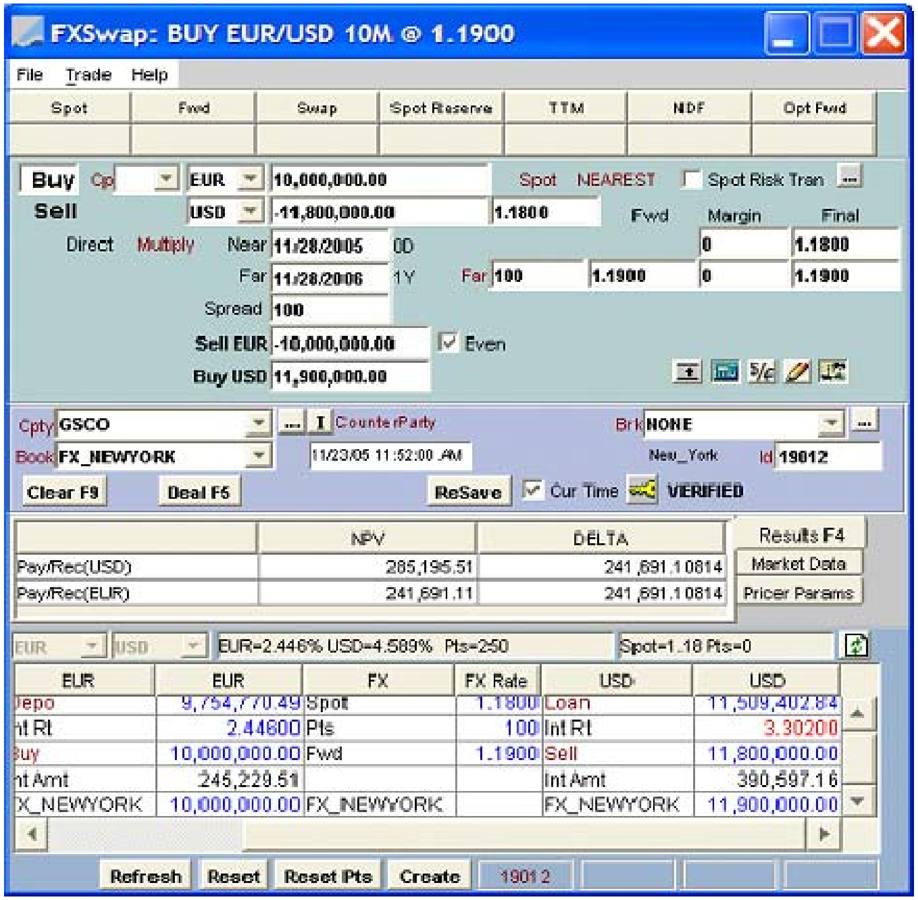
5.2 FX Swaps

FX Swap trades are valued using **PricerFXSwap**, which in fact does nothing but delegate the computation to the spot and forward pricers. Thus:

Whatever pricers are specified in PricerConfig for the FX and FXForward trades will be used for FX Swap trades.

One can use PricerFX for spot and PricerFXForward for forwards; alternately one can use PricerFXForwardHomeBased for both spot and forward trades. Having configured these, the valuing of FX swap trades follows as a consequence.

Following is a pricing example.



FX Swap Valuation

Value Date	8/11/2006							
			Near Leg	**Forward Leg**				
Primary Ccy	EUR	**Date**	11/28/2005	11/28/2006				
Quoting Ccy	USD	**Primary Amt(EUR)**	10,000,000.00	(10,000,000.00)				
FX Rate	1.18	**Quoting Amt(USD)**	(11,800,000.00)	11,900,000.00				
		FX Rate	1.18	1.19				
		Zero (EUR)	2.44%	2.44%		2.44%		
		Zero(USD)	4.59%	4.59%		4.59%		
		Time(yrs)	-0.7014	0.2986				
		Theo Fwd FX	1.1622	1.1876				
		Theo Points	-177.52	75.58				
		Actual Points	0	100	**Net(Native)**	**Net(EUR)**	**Net(USD)**	
		NPV (EUR)	$ 10,172,809.90	$ (9,927,314.76)	$ 245,495.14	$ 245,495.14	$ 289,684.27	
		NPV (USD)	$ (12,185,864.52)	$ 11,738,076.90	$ (447,787.62)	$ (379,481.03)	$ (447,787.62)	
		NPV (Net)				$ (133,985.89)	$ (158,103.35)	
7.389056099	20.085537	**PV01(EUR)**	713.52	296.46				
0.367879441		**PV01(USD)**	(854.71)	(350.53)				
						Forward	Spot	
148.4131591					**Pay/Rec**	11,514,804.49	11,800,000.00	285,195.51
			$ 12,003,915.69	$ (11,714,231.41)	**Pay/Rec(EUR)**	9,758,308.89	10,000,000.00	241,691.11
			$ (181,948.83)	$ 23,845.48				
				$ 11,714,231.41				

5.3 FX Cash

The FX Cash products specify a series of FX Forward trades. The NPV of the FX Cash product is just the sum of the NPVs of the FX Forward trades. Thus in pricing FX Cash trades the same methodology as pricing FX Forward trades is used.

Two pricers are available, depending on which method of valuing an FX Forward is desired. **PricerFXCash** will use the methodology of PricerFXForward, while **PricerFXCashHomeBased** will use the methodology of PricerFXForwardHomeBased.

5.4 FX NDF

The net present value of the FX NDF trade is the same as pricing an FX trade before the reset price is set and before reset date time.

After reset date & time and after the reset price is set, if the settlement is in quoting currency, the value in quoting currency is primary amount * (reset price – trade price), which is then discounted to valuation date using the quoting discount curve. If the settlement is in primary currency, the value is the primary amount - (primary amount * trade price / reset price), which is then converted to quoting currency using the forward price, and discounted to valuation date using the quoting discount curve.

5.5 FX Option Forward

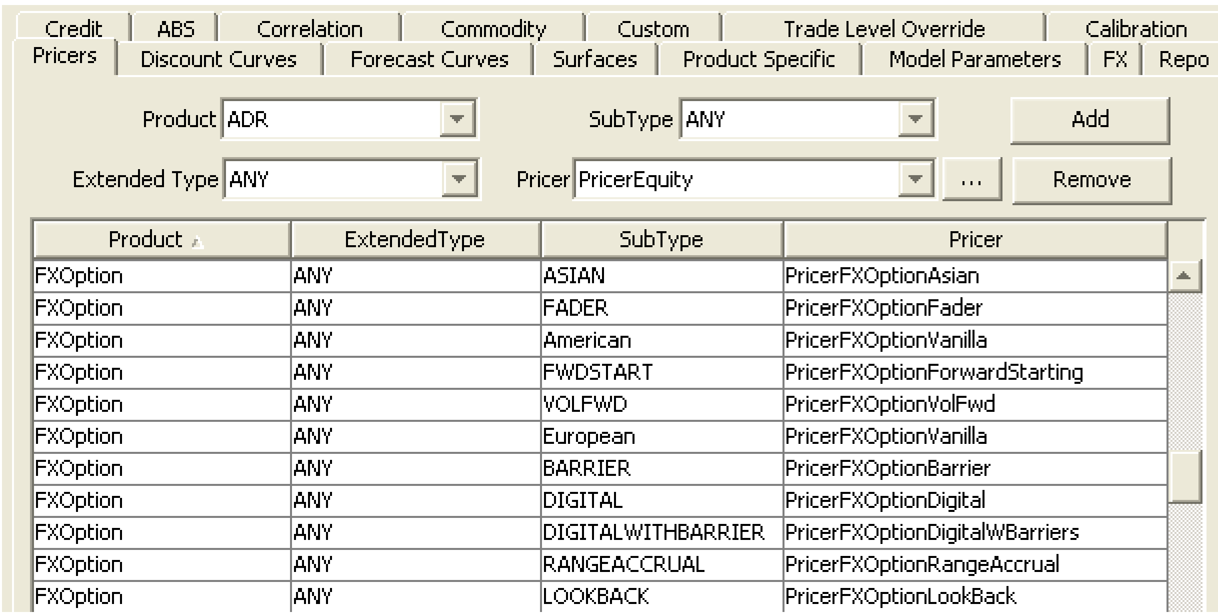
The FX Option Forward product gives discretion to one of the trade parties to convert currency at any time within a specified period. The pricers for this perform a simple computation using the user's input for a projected settle date. On that settle date it is assumed that any outstanding balance will be converted. Thus the FX Option Forward is converted to a simple FX Forward trade for pricing purposes.

How this effective FX Forward trade is priced depends on which of the two pricers one has chosen. **PricerFXOptionForward** will use the methodology of PricerFXForward, while **PricerFXOptionForwardHomeBased** will use the methodology of PricerFXForwardHomeBased.

Section 6. FX Option Valuation

6.1 Overview

In the pricer configuration, the following pricers should be specified for FX Options.



Product	ExtendedType	SubType	Pricer
FXOption	ANY	ASIAN	PricerFXOptionAsian
FXOption	ANY	FADER	PricerFXOptionFader
FXOption	ANY	American	PricerFXOptionVanilla
FXOption	ANY	FWDSTART	PricerFXOptionForwardStarting
FXOption	ANY	VOLFWD	PricerFXOptionVolFwd
FXOption	ANY	European	PricerFXOptionVanilla
FXOption	ANY	BARRIER	PricerFXOptionBarrier
FXOption	ANY	DIGITAL	PricerFXOptionDigital
FXOption	ANY	DIGITALWITHBARRIER	PricerFXOptionDigitalWBarriers
FXOption	ANY	RANGEACCRUAL	PricerFXOptionRangeAccrual
FXOption	ANY	LOOKBACK	PricerFXOptionLookBack

If the subtypes indicated above are not available for selection, you can add them to the domain "FXOption.subtype".

When pricing partial double barriers, and partial digitals with barrier, you can set the pricing parameter VALUE_NON_STD_DBL_BARRIER to select the pricing model:

- If True, value strictly partial double barriers using the Quad implementation of Black Scholes dynamics.

- If False, value only a combination of start-of-period, end-of-period, at-expiry partial double barriers.

PricerFXOptionBarrierMixture and PricerFXOptionDigitalMixture

For BARRIER options, you can also use PricerFXOptionBarrierMixture, and for DIGITAL options you can also use PricerFXOptionDigitalMixture. To calculate numerical Greeks, the model will not re-calibrate unless the pricing parameter MIXTURE_CALIBRATE_FOR_GREEKS is set to true (default is false).

These pricers use the Mixture Model Calibration: Given a market volatility surface of Black-Scholes volatilities with a delta strike axis, MixtureModelCalibration creates a set of local volatility surfaces and solves for a corresponding set of weights. The weights minimize the errors in the set of equations **C(Dj) = Sum_i w(i) * Cnum(Dj, Si)**, where:

- C(Dj) is the Black-Scholes call price for the jth Delta, using the market surface
- Cnum(Dj, Si) is the call price for the jth Delta evaluated with a numerical method that uses the ith local volatility surface
- w(i) is the ith weight, to be solved for, subject to the constraints: 0 <= w(i) <= 1 and Sum_i w(i) = 1.

In evaluating the calls for each delta Dj, the strike is first solved for that reproduces that delta using the market volatility. Then the Black-Scholes and the numerical routine evaluate the call option that has that strike.

PricerFXOptionForwardStarting

For Forward Starting options, the volatility that is entered in the pricing sheet and trade window, is the volatility associated from start fixing date to option expiry date. If the volatility has not been entered, the volatility is calculated by:

forward volatility = sqr_root((1/(T-t))*(T*volAtExpiry^2 - t*volAtStartDate^2))

where:

- T = time in years from spot date to expiry date
- t = time in years from spot date to start fixing date
- volAtExpiry = the implied volatility given delta of the vanilla option from today to expiry using the effective strike
- volAtStartDate = the implied volatility given delta of the vanilla option from today to start fixing date using the effective strike

The interest rates that are entered in the pricing sheet and trade window, are the interest rates from spot date to delivery date. The interest rate from spot date to the start fixing date is retrieved from the curves and currently the user cannot enter this rate. The forward interest rate from start fixing date to expiry date is calculated by:

interest rate = (T*rExpiry-t*rStart) / (T-t)

where:

- T = time in years from spot date to delivery date
- t = time in years from spot date to spot date of the start fixing date
- rExpiry = continuous interest rate from spot date to delivery date (can be entered by user)
- rStart = continuous interest rate from spot to spot date of the start fixing date (has to be retrieved from curve)

PricerFXOptionDigitalWBarriers

For a Down and Out, Up and In where spot is above the up barrier, type A will price equivalent to the underlying digital whereas type B will price equivalent to a down and out.

Same methodology for a Down and In, Up and Out where the spot is below the down barrier.

Intraday Options

Pricing parameter VALUE_INTRADAY_OPTIONS controls the valuation of intraday options.

- If True, on the expiry date of the option or barrier end date for partial barrier options, the pricer measures will evaluate the option based on units of time in milliseconds to option expiry or barrier expiry

- If False, on the expiry date of the option or barrier, the pricer measures will reflect the option as expired or barrier expired.

6.2 FX Option Pricers

FX Option pricers are provided to value vanilla and exotic options within the Black-Scholes model of the FX rate environment. The valuation algorithms called by these pricers are listed in the following table.

For barrier options, a Vanna-Volga adjustment can be added to the Black-Scholes price. Please see the section on the Vanna-Volga Method for more details.

A **Heston model** pricer for barrier options is also provided. This is described in the Heston section.

The Black-Scholes volatility used for each option is shown in the table. In looking up volatilities for exotics, a vanilla European option is created with the same expiry and strike as the given option. The indicated volatility for this vanilla option is then looked up from the volatility surface. For most exotic options, only the ATM volatility of the created vanilla option is used, that is, the volatility from the surface at a forward delta of 50. For options that have a representation as the sum of vanilla options, such as barrier-at-expiry options, volatilities are found for each of those constituents.

Page references are to the following works:

Haug: *The Complete Guide to Option Pricing Formulas*, Espen Gaarder Haug (McGraw-Hill, 1998)

Zhang: *Exotic Options*, Peter Zhang (World Scientific, 1997)

Pelsser: "Pricing Double Barrier Options: An Analytical Approach," Antoon Pelsser (ABN-Amro, 1997)

Andricopoulos: "Universal option valuation using quadrature methods," Ari D. Andricopoulos, Martin Widdicks, Peter W. Duck, David P. Newton (Journal of Financial Economics 67 (2003) 447–471)

Option Type	Model (Black-Scholes FX environment)	Black-Scholes Volatility
European Vanilla	Garman-Kohlhagen (Haug, p. 6)	Vanilla volatility
American Vanilla	Garman-Kohlhagen, binomial tree implementation	Vanilla volatility
Single K/I or K/O	Reiner-Rubinstein (Haug p. 70 ff)	ATM; Vanna-Volga adjustment
Single K/I-At Expiry or K/O-At Expiry	Replication by European Vanillas and Digitals	Multiple volatilities
Digital Cash-or-Nothing At Expiry	Reiner-Rubinstein (Haug, p. 88)	Vanilla volatility
Partial 1-Day Barrier (single, double, KIKO)	Brockhaus, Ferraris (Haug, 2nd Ed, p. 122)	ATM
Partial Single Barrier (Early-start, Late-Ending)	Heynen and Kat (Haug, p. 77)	ATM
Window Single Barrier	Armstrong	ATM
Digital Barrier One Touch Cash-or-Nothing, Instant or Expiry payout	Reiner-Rubinstein (Haug, p. 92 ff)	ATM
Digital Barrier No Touch Cash-or-Nothing	Reiner-Rubinstein (Haug, p. 92 ff)	ATM
Digital Asset-or-Nothing At Expiry	Reiner-Rubinstein (Haug, p. 90)	Vanilla volatility
Digital Barrier One Touch Asset-or-Nothing, Instant or Expiry payout	Reiner-Rubinstein (Haug, p. 92 ff)	ATM
Digital Barrier No Touch Cash-or-Nothing	Reiner-Rubinstein (Haug, p. 92 ff)	ATM
Double Barrier One Touch K/I or K/O	Ikeda-Kunitomo (Haug, p. 72 ff)	ATM; Vanna-Volga adjustment
Double K/I-At Expiry or K/O-At Expiry	Replication by European Vanillas and Digitals	Multiple volatilities
Double Barrier K/I & K/O Type B	Replication with single barrier and double one-touch barrier options	ATM
Digital Double Barrier One Touch Cash-or-Nothing or Asset-or-Nothing	Pelsser	ATM
Digital Double Barrier No touch Cash-or-Nothing or Asset-or-Nothing	Pelsser	ATM
Asian Average Rate, Arithmetic Average	Turnbull-Wakeman (Haug, p. 97)	ATM

Option Type	Model (Black-Scholes FX environment)	Black-Scholes Volatility
Asian Average Strike, Arithmetic Average	Zhang (p. 140 ff)	ATM
Asian Average Rate, Geometric Average	Kemna-Vorst/Zhang (Zhang, p. 117)	ATM
Asian Average Strike, Geometric Average	Zhang (p. 124 ff)	ATM
Look Back Floating Rate	Goldman (Haug, p. 61)	ATM
Look Back Floating Strike	Conze-Viswanathan (Haug, p. 63)	ATM
Compound FX Option	Geske, implemented in PricerFXCompoundOption (Haug, p. 43)	Vanilla volatility
Digital with Barrier (Single KI/KO)	Reiner-Rubinstein (Haug, p. 92 ff)	ATM
Digital with Barrier Single KI, Double knock-in, KIKO	Replication with single digital barrier and double KO digital barrier options	ATM
Range Accrual (Above, Below, In, Out)	Replication of Digital Cash-or-Nothing At Expiry	ATM
Range Accrual Single and Double Barrier	Replication of digital with single barrier	ATM
Range Accrual Double Barrier	Replication of Digital with double Barrier	ATM
Forward Starting	Zhang (p. 85 ff)	ATM
Single & Double Barrier Options (Full Window or Partial Window)	Andricopoulos	Mixture Volatility Model
Digital Cash-or-Nothing, Expiry Payout (one touch, no touch, double binary)	Andricopoulos	Mixture Volatility Model

For European vanilla options and simple digitals, these are computed analytically using the Black model. For all other options, they are computed numerically by shifting the underlying parameters.

DELTA

The change in option premium with respect to the change in the spot rate. Let Q(S) denote the premium of the option at spot rate S when the premium is in quoting currency. Then the default definition is

$$DELTA_Q = \frac{Q(S+dS) - Q(S-dS)}{2dS}$$

For vanilla or simple digital options, this value is computed analytically in the limit as dS goes to zero. For other exotics, this is calculated numerically with dS equal to 0.00001. The subscript “Q” is used here to show that premium is in quoting currency.

By default, DELTA is defined with the option premium assumed to be in quoting currency; then DELTA is in primary currency. However, if the premium is in primary currency, the sensitivity with respect to spot must be computed differently. Let P(S) be the premium when expressed in primary currency. That is:

$$P(S) = Q(S)/S$$

assuming S is defined in terms of quoting currency per one unit of primary currency. Then the sensitivity of P(S) with reset to changes in S is

$$\frac{P(S+dS)-P(S-dS)}{2dS} = \frac{Q(S+dS)/(S+dS)-Q(S-dS)/(S-dS)}{2dS}$$
$$\approx \frac{1}{S}[\frac{Q(S+dS)-Q(S-dS)}{2dS} - \frac{Q(S)}{S}]$$
$$= \frac{1}{S}(DELTA_Q - P(S))$$

The sensitivity when premium is in primary currency is therefore “delta minus premium” divided by the spot rate.

By setting the parameter USE_DELTA_TERM, the meaning of DELTA can be changed to reflect the premium currency. One must first have specified the “Delta Term” of the currency pair to define the currency used for option premium (mark-to-market currency). Then one must set the pricer parameter USE_DELTA_TERM to true. The behavior is summarized as follows, with the notation as defined above.

If USE_DELTA_TERM is False (the default):

DELTA = $DELTA_Q$

If USE_DELTA_TERM is True:

If premium is in quoting currency:

DELTA = $DELTA_Q$

If premium is in primary currency:

DELTA = $DELTA_Q$ – P(S)

In all cases, DELTA is in primary currency.

The use of the parameter can be avoided by using the measure HEDGE_DELTA instead of DELTA.

See also:

- DELTA_W_PREMIUM
- HEDGE_DELTA
- FWD_DELTA

DELTA_W_PREMIUM

The DELTA of the option, if premium were in quoting currency, minus the premium of the option in primary currency. See the discussion under DELTA. In terms of the notation used there,

- DELTA_W_PREMIUM = $DELTA_Q$ – P(S).

It is always in primary currency.

DDELTA_DVOL

The cross-derivative dDelta/dVol of price with respect to the spot rate and the volatility; also known as dVega/dSpot, *stability ratio*, and *vanna*.

DVEGA_DSPOT

The cross-derivative (dVega/dSpot) of option price with respect to the spot rate and the volatility. Although in the continuous limit this is theoretically the same as DDELTA_DVOL, numerical implementations can lead to different results.

DVEGA_DVOL

The change of Vega with volatility, which is the second derivative of option price with respect to volatility; also known as *vomma* or *kappa*.

FWD_DELTA

The change in forward NPV with respect to the forward rate. Forward NPV is spot NPV divided by the quoting currency discount rate between spot and the expiry date. One can show that the Forward Delta is related to the spot Delta by

$$FWD\_DELTA = \frac{DELTA}{DF_{Pmry}(t_{spot}, t_{\exp})} .$$

The denominator is the discount factor from spot to expiry for the primary currency. If Delta was valued on today's date rather than the spot date, one must first move Delta to the spot date by dividing by the quoting currency discount factor from today to spot.

For a vanilla European option, in terms of the usual Black-Scholes variables,

$$FWD\_DELTA = N(d_1) .$$

GAMMA

The change in Delta with respect to changes in the spot rate. Let Q(S) denote the premium of the option at spot rate S when the premium is in quoting currency. Then Gamma is computed empirically as:

$$GAMMA = \frac{Q(S + dS) + Q(S - dS) - 2Q(S)}{(dS)^2} .$$

For vanilla or simple digital options, this value is computed analytically in the limit as dS goes to zero. For other exotics, this is calculated numerically with dS equal to 0.00001.

HEDGE_DELTA

The sensitivity of the option premium to the underlying spot rate, taking into account the option premium currency, with the resulting sensitivity expressed in primary currency. Using the same notation found under the description of DELTA, the measure has the following behavior

If premium is in quoting currency:

HEDGE_DELTA = $DELTA_Q$

If premium is in primary currency:

HEDGE_DELTA = $DELTA_Q$ – P(S)

MOD_DELTA

The total derivative of option price with respect to the spot rate, taking into account dependence of volatility on the spot rate. Calculated numerically as a two-sided delta by shifting the spot rate by a relative factor of 1.00001 and looking up the volatility at the shifted and unshifted values of the spot rate.

MOD_GAMMA

The total derivative of MOD_DELTA with respect to the spot rate, taking into account dependence of volatility on the spot rate. Computed in a similar way to MOD_DELTA.

MOD_VEGA

The total derivative of price with respect to at-the-money volatility, taking into account shifts of the volatility surface as the ATM volatility changes, assuming the skew remains unchanged.

NPV

Premium amount. Can include fees or not depending on the INCLUDE_FEES parameter.

PRICE

The unsigned premium amount per unit notional. Fees are not included.

PV

Premium amount (NPV) without fees.

REAL_THETA

The change of option premium with time taking into account the time dependence of the volatility surface and interest curves. Computed numerically by moving time one day in the future, finding the volatility and interest rates associated with valuation on that date, recalculating with those values and taking the difference.

RHO

Change in NPV (option premium) for absolute 1% (100 bp) change in interest rate of primary currency.

RHO2

Change in NPV (option premium) for absolute 1% (100 bp) change in interest rate of quoting currency.

THETA

Change in option premium with respect to change in the remaining time of the option. For vanilla and digital options, the analytic Black-Scholes theta; otherwise computed numerically using a one day change in the remaining time.

THETA2

Change in option premium with respect to change in the remaining time of the option, where the forward rate is held fixed. Computed numerically using a one day change in the remaining time. To keep the forward rate constant, the spot rate is shifted forward one day using the quoting currency and primary currency interest rates:

$$S(t+1) = S(t)e^{(r_Q - r_P)/365}$$

The revaluation one day in the future is done with the advanced spot rate S(t+1).

VEGA

Change in option premium with respect to change in volatility. For quoting-currency premium Q(S,σ) at spot rate S and volatility σ,

$$VEGA = \frac{Q(S,\sigma + d\sigma) - Q(S,\sigma)}{d\sigma} \quad .$$

Section 7. Option Algorithms

This chapter described the models used for pricing.

References

Haug: *The Complete Guide to Option Pricing Formulas*, Espen Gaarder Haug (McGraw-Hill, 1998)

Zhang: *Exotic Options*, Peter Zhang (World Scientific, 1997)

Pelsser: "Pricing Double Barrier Options: An Analytical Approach," Antoon Pelsser (ABN-Amro, 1997)

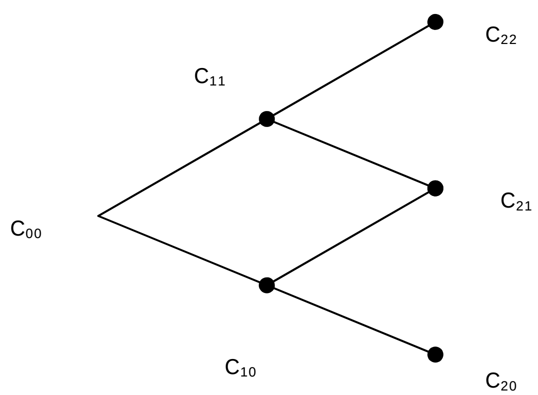
7.1 Garman-Kohlhagen

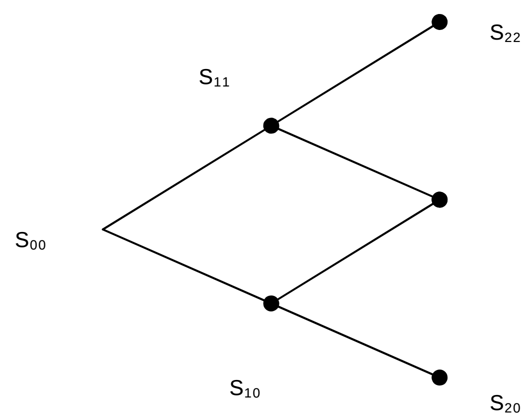
This is a version of Black-Scholes used to price European style FX options.

7.2 Binomial Single Asset Tree

Multi-exercise single asset vanilla options: for FX options, equity options, and future options.

The standard Cox, Ross and Rubinstein version of the Binomial tree model is used to value American-style options. The Binomial Tree model is a discrete version of the continuous Black-Scholes model and discretizes time into equal intervals. At the time t=0, the path of the security can either go up exp(u) or down exp(d) with a probabilities of p_{up} and p_{down}, respectively. A tree of asset prices can then be constructed and the option payoff at each node can be calculated and discounted back to time t=0 to arrive at the option value. The equations for constructing the tree are shown below.





$$u = \exp(\sigma\sqrt{\Delta t_e})$$

$$d = \frac{1}{u}$$

$$p_{up} = \frac{\exp((r_g - q)\Delta t_e) - d}{u - d}$$

$$p_{down} = 1 - p_{up}$$

The American-style option price must be calculated at each node. This is done using the following equations:

At maturity (t = T), the option price is computed as follows:

$$C = \max[S - K, 0]$$

$$P = \max[K - S, 0]$$

Prior to maturity, the American option price is computed as follows:

$$C_{11} = \max[S_{11} - K,\ e^{-r_p t_p}(p_{up} * C_{22} + p_{down} C_{21})]$$

$$P_{11} = \max[K - S_{11},\ e^{-r_p t_p}(p_{up} * P_{22} + p_{down} P_{21})]$$

The variables are described as follows:

- S is the spot price of the underlying security.
- K is the strike price
- r_g, r_p are equal to the continuous discount rates (see table on page 5 for more information)
- q is the continuous dividend yield of the security

  Note that both r and q are continuous rates. Typically, the market quotes periodically compounded rates which can be changed to continuous rates using the following equation:

  $r_{continuous} = [(\ln(r_{periodic} * t + 1)] / t$
- σ is the volatility of returns of the underlying security
- t_e is the time period from the valuation date to the option's expiration date
- t_p is the time period from the spot date (typically 2 days after the transaction date) and the settlement date (typically 2 days after expiration)

The price of the option is affected by changes in the various inputs. The change in the option price can be determined by using the formulas shown below.

Delta

Delta is the rate of change of the option value with respect to the underlying's spot price.

$$\Delta_c = \frac{\partial C}{\partial S} = \frac{C_{11} - C_{10}}{S_{00}(u - d)}$$

$$\Delta_p = \frac{\partial P}{\partial S} = \frac{P_{11} - P_{10}}{S_{00}(u - d)}$$

Delta Premium

$$\Delta_{c,premium} = \frac{C_{11} - C_{10}}{S_{00}(u - d)} - C$$

$$\Delta_{p,premium} = \frac{P_{11} - P_{10}}{S_{00}(u - d)} + P$$

Delta Forward

$$\Delta_c = \frac{C_{11} - C_{10}}{S_{00} * e^{t_e(r_g - q)} * (u - d)}$$

$$\Delta_p = \frac{\partial P}{\partial S} = \frac{P_{11} - P_{10}}{S_{00} * e^{t_e(r_g - q)} * (u-d)}$$

Gamma

Gamma represents the rate of change of delta with respect to the spot price. It can be computed as the second derivative of the option premium with respect to spot, or the first derivative of the option's delta. Delta can also be computed with respect to a 1% change in the underlying.

$$\Gamma_c = \frac{\partial^2 C}{\partial S^2} = \frac{\dfrac{C_{22} - C_{21}}{S_{00}(u^2 - 1)} - \dfrac{C_{21} - C_{20}}{S_{00}(1 - d^2)}}{.5(S_{00}u^2 - S_{00}d^2)}$$

$$\Gamma_p = \Gamma_c$$

Theta

Theta measures the sensitivity of the option price with respect to time.

$$\Theta_c = -\frac{\partial C}{\partial \tau} = -\frac{C_{21} - C_{00}}{2\Delta\tau}$$

$$\Theta_p = -\frac{\partial P}{\partial \tau} = -\frac{P_{21} - P_{00}}{2\Delta\tau}$$

Vega

Vega measures the rate of change of the option value with respect to changes in the volatility of the underlying.

$$\nu_c = \frac{\partial C}{\partial \sigma} = \frac{C(\sigma + \Delta\sigma) - C(\sigma - \Delta\sigma)}{2\Delta\sigma}$$

$$\nu_p = \frac{\partial P}{\partial \sigma} = \nu_c$$

rho

rho measures the rate of change of the option value with respect to the risk-free rate.

$$\rho_c = \frac{\partial C}{\partial r} = \frac{C(r + \Delta r) - C(r - \Delta r)}{2\Delta r}$$

$$\rho_p = \frac{\partial P}{\partial r} = \frac{P(r + \Delta r) - P(r - \Delta r)}{2\Delta r}$$

rho$_2$

rho$_2$ is the sensitivity of the option price to dividend yield.

$$\rho_{2,c} = \frac{\partial C}{\partial q} = \frac{C(q + \Delta q) - C(q - \Delta q)}{2\Delta q}$$

$$\rho_{2,p} = \frac{\partial P}{\partial q} = \frac{P(q + \Delta q) - P(q - \Delta q)}{2\Delta q}$$

7.3 Binary/Digital Options

7.3.1 Cash or Nothing Call Option

NPV

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

$$d_2 = \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{\sigma\sqrt{t_e}} - \frac{\sigma\sqrt{t_e}}{2}$$

Delta

$$\Delta_c = \frac{\partial C}{\partial S} = Ke^{-r_p t_p} n(d_2) \frac{\partial d_2}{\partial S}$$

$$\frac{\partial d_2}{\partial S} = \frac{\partial}{\partial S} \frac{\ln S - \ln K}{\sigma\sqrt{t_e}} = \frac{1}{S\sigma\sqrt{t_e}}$$

$$\Delta_c = \frac{Ke^{-r_p t_p} n(d_2)}{S\sigma\sqrt{t_e}}$$

Delta Forward

$$\Delta_{c,\,forward} = \frac{\partial C}{\partial F} = Ke^{-r_p t_p} n(d_2) \frac{\partial d_2}{\partial F}$$

$$\frac{\partial d_2}{\partial F} = \frac{\partial}{\partial F} \frac{\ln F - \ln K}{\sigma\sqrt{t_e}} = \frac{1}{F\sigma\sqrt{t_e}}$$

$$\Delta_{c,\,forward} = \frac{Ke^{-r_p t_p - (r_g - q)t_e} n(d_2)}{S\sigma\sqrt{t_e}}$$

Gamma

$$\Gamma_c = \frac{\partial^2 C}{\partial S^2} = \frac{\partial}{\partial S} \frac{Ke^{-r_p t_p} n(d_2)}{S\sigma\sqrt{t_e}}$$

$$\frac{\partial n(d_2)}{\partial S} = \frac{\partial}{\partial S} \frac{-n(d_2) * d_2}{S\sigma\sqrt{t_e}}$$

Combining the above equations gives :

$$\Gamma_c = \frac{e^{-r_p t_p} Kn(d_2)}{\sigma\sqrt{t_e}} \left(\frac{-n(d_2) * d_2}{S^2\sigma\sqrt{t_e}} - \frac{n(d_2)}{S^2} \right)$$

$$\Gamma_c = -\frac{e^{-r_p t_p} Kn(d_2)}{S^2\sigma\sqrt{t_e}} \left(\frac{d_2}{\sigma\sqrt{t_e}} + 1 \right)$$

Vega

$$\nu_c = \frac{\partial C}{\partial \sigma} = e^{-r_p t_p} Kn(d_2) \frac{\partial d_2}{\partial \sigma}$$

$$\frac{\partial d_2}{\partial \sigma} = -\frac{\ln\left[\dfrac{Se^{-qt_e}}{Ke^{-r_g t_e}} \right]}{\sigma^2\sqrt{t_e}} - \frac{\sqrt{t_e}}{2}$$

$$\nu_c = -Ke^{-r_p t_p} n(d_2) \left(\frac{\ln\left[\dfrac{Se^{-qt_e}}{Ke^{-r_g t_e}} \right]}{\sigma^2\sqrt{t_e}} + \frac{\sqrt{t_e}}{2} \right)$$

rho$_{r_p}$

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

rho$_{r_g}$

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

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

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

Substituting equation (3) gives:

$$\rho_{c,g} = t_e e^{(r_g - q)t_e - r_p t_p} SN(d_1) + e^{(r_g - q)t_e - r_p t_p} Sn(d_1) \frac{\sqrt{t_e}}{\sigma}$$

rho$_{\mathbf{q}}$

$$\rho_{2,c} = \frac{\partial C}{\partial q} = -t_e e^{(r_g - q)t_e - r_p t_p} SN(d_1) + e^{(r_g - q)t_e - r_p t_p} Sn(d_1) \frac{\partial d_1}{\partial q}$$

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

$$\frac{\partial d_1}{\partial q} = \frac{-\sqrt{t_e}}{\sigma}$$

Combining the above equations gives:

$$\rho_{2,c} = -t_e e^{(r_g - q)t_e - r_p t_p} SN(d_1) - e^{(r_g - q)t_e - r_p t_p} Sn(d_1) \frac{\sqrt{t_e}}{\sigma}$$

Theta

Defining $t_1 = \tau - T_1, t_2 = \tau - T_2$, and $t_3 = \tau - T_3$:

$$\Theta_c = -\frac{\partial C}{\partial \tau} = Ke^{-r_p t_p}\left(-n(d_2)\frac{\partial d_2}{\partial \tau} + r_p N(d_2)\right)$$

$$\frac{\partial d_2}{\partial \tau} = \frac{r_g - q}{\sigma\sqrt{t_e}} - \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{2\sigma t_e^{1.5}} - \frac{\sigma}{4\sqrt{t_e}}$$

Combining the above equations gives :

$$\Theta_c = -e^{-r_p t_p}K\left\{n(d_2)\left(\frac{r_g - q}{\sigma\sqrt{t_e}} - \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{2\sigma t_e^{1.5}} - \frac{\sigma}{4\sqrt{t_e}}\right) - r_p N(d_2)\right\}$$

7.3.2 Cash or Nothing Put Options

NPV

$$P = Ke^{-r_p t_p} N(-d_2)$$

$$d_2 = \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{\sigma\sqrt{t_e}} - \frac{\sigma\sqrt{t_e}}{2}$$

$$n(d_2) = \frac{e^{\frac{-d_2^2}{2}}}{\sqrt{2\pi}}$$

Delta

$$\Delta_p = \frac{\partial P}{\partial S} = Ke^{-r_p t_p} n(-d_2)\frac{\partial - d_2}{\partial S}$$

$$\frac{\partial d_2}{\partial S} = \frac{\partial}{\partial S}\frac{\ln S - \ln K}{\sigma\sqrt{t_e}} = \frac{1}{S\sigma\sqrt{t_e}}$$

$$\Delta_p = -\frac{Ke^{-r_p t_p} n(-d_2)}{S\sigma\sqrt{t_e}}$$

Delta Forward

$$\Delta_{p,\,forward} = \frac{\partial P}{\partial F} = Ke^{-r_p t_p} n(-d_2) \frac{\partial - d_2}{\partial F}$$

$$\frac{\partial d_2}{\partial F} = \frac{\partial}{\partial F} \frac{\ln F - \ln K}{\sigma\sqrt{t_e}} = \frac{1}{F\sigma\sqrt{t_e}}$$

$$\Delta_{p,\,forward} = -\frac{Ke^{-r_p t_p - (r_g - q)t_e} n(-d_2)}{S\sigma\sqrt{t_e}}$$

Gamma

$$\Gamma_p = \frac{\partial^2 P}{\partial S^2} = \frac{\partial}{\partial S} - \frac{Ke^{-r_p t_p} n(-d_2)}{S\sigma\sqrt{t_e}}$$

$$\frac{\partial d_2}{\partial S} = \frac{\partial}{\partial S} \frac{-n(d_2) * d_2}{S\sigma\sqrt{t_e}}$$

Combining the above equations gives :

$$\Gamma_p = \frac{e^{-rt_2} Kn(d_2)}{\sigma\sqrt{t_e}} \left(\frac{n(d_2) * d_2}{S^2 \sigma\sqrt{t_e}} + \frac{n(d_2)}{S^2} \right)$$

$$\Gamma_p = \frac{e^{-r_p t_p} Kn(d_2)}{S^2 \sigma\sqrt{t_e}} \left(\frac{d_2}{\sigma\sqrt{t_e}} + 1 \right)$$

Vega

$$\nu_p = \frac{\partial P}{\partial \sigma} = e^{-r_p t_p} Kn(d_2) \frac{\partial - d_2}{\partial \sigma}$$

$$\frac{\partial d_2}{\partial \sigma} = -\frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{\sigma^2\sqrt{t_e}} - \frac{\sqrt{t_e}}{2}$$

$$\nu_p = Ke^{-r_p t_p} n(d_2) \left(\frac{\ln \frac{Se^{-qt_e}}{Ke^{-r_g t_e}}}{\sigma^2\sqrt{t_e}} + \frac{\sqrt{t_e}}{2} \right)$$

$\mathbf{rho_{r_p}}$

$$\rho_{p,p} = \frac{\partial P}{\partial r_p} = -t_p e^{(r_g - q)t_e - r_p t_p} SN(-d_1)$$

rho$_{r_g}$

$$\rho_{g,p} = \frac{\partial P}{\partial r_g} = t_e e^{(r_g - q)t_e - r_p t_p} SN(-d_1) + e^{(r_g - q)t_e - r_p t_p} Sn(-d_1) \frac{\partial - d_1}{\partial r_g}$$

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

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

Combining equations gives :

$$\rho_{g,p} = t_e e^{(r_g - q)t_e - r_p t_p} SN(-d_1) - e^{(r_g - q)t_e - r_p t_p} Sn(-d_1) \frac{\sqrt{t_e}}{\sigma}$$

rho$_q$

$$\rho_{2,p} = \frac{\partial P}{\partial q} = -t_e e^{(r_g - q)t_e - r_p t_p} SN(-d_1) + e^{(r_g - q)t_e - r_p t_p} Sn(-d_1) \frac{\partial - d_1}{\partial r_g}$$

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

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

Combining equations gives :

$$\rho_{g,p} = -t_e e^{(r_g - q)t_e - r_p t_p} SN(-d_1) + e^{(r_g - q)t_e - r_p t_p} Sn(-d_1) \frac{\sqrt{t_e}}{\sigma}$$

Theta

$$\text{Defining } t_1 = \tau - T_1, t_2 = \tau - T_2, \text{and } t_3 = \tau - T_3:$$

$$\Theta_p = -\frac{\partial P}{\partial \tau} = -e^{-r_p t_p} K\left(n(d_2)\frac{\partial - d_2}{\partial \tau} - r_p N(-d_2) \right)$$

$$\frac{\partial d_2}{\partial \tau} = \frac{r_g - q}{\sigma\sqrt{t_e}} - \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{2\sigma t_e^{1.5}} - \frac{\sigma}{4\sqrt{t_e}}$$

Combining the above equations gives:

$$\Theta_p = -e^{-r_p t_p} K\left\{ n(d_2)\left(\frac{r_g - q}{\sigma\sqrt{t_e}} - \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{2\sigma t_e^{1.5}} - \frac{\sigma}{4\sqrt{t_e}} \right) - r_p N(-d_2) \right\}$$

7.3.3 Asset or Nothing Binary Call Option

NPV

$$C = Se^{(r_g - q)t_e - r_p t_p} N(d_1)$$

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

$$n(d_1) = \frac{e^{\frac{-d_1^2}{2}}}{\sqrt{2\pi}}$$

Delta

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

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

Combining the equations above gives:

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

Delta Forward

$$\Delta_{c,\,forward} = \frac{\partial C}{\partial F} = e^{-r_p t_p}\left(Fn(d_1)\frac{\partial d_1}{\partial S} + N(d_1) \right)$$

$$\frac{\partial d_1}{\partial F} = \frac{\partial}{\partial S}\frac{\ln F - \ln K}{\sigma\sqrt{t_e}} = \frac{1}{F\sigma\sqrt{t_e}}$$

Combining the equations above gives:

$$\Delta_{c,\,forward} = e^{-r_p t_p}\left(N(d_1) + \frac{n(d_1)}{\sigma\sqrt{t_e}} \right)$$

Gamma

$$\Gamma_c = \frac{\partial^2 C}{\partial S^2} = \frac{\partial}{\partial S} e^{(r_g - q)t_e - r_p t_p}\left(N(d_1) + \frac{n(d_1)}{\sigma\sqrt{t_e}} \right)$$

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

The partial derivatives are calculated as follows:

$$\frac{\partial d_1}{\partial S} = \frac{\partial}{\partial S}\frac{(\ln Se^{-qt_e} - \ln Ke^{-r_g t_e})}{\sigma\sqrt{t_1}} = \frac{1}{S\sigma\sqrt{t_e}}$$

$$\frac{\partial n(d_1)}{\partial S} = \frac{n(d_1) * -d_1}{S\sigma\sqrt{t_e}}$$

Combining the above equations gives:

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

Vega

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

$$\frac{\partial d_1}{\partial \sigma} = -\frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{\sigma^2 \sqrt{t_e}} + \frac{\sqrt{t_e}}{2}$$

Combining the above equations gives:

$$\nu_c = e^{(r_g - q)t_e - r_p t_p} Sn(d_1) \left(-\frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{\sigma^2 \sqrt{t_e}} + \frac{\sqrt{t_e}}{2} \right)$$

rho$_{r_p}$

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

rho$_{r_g}$

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

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

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

Substituting equation (3) gives:

$$\rho_{c,g} = \frac{e^{(r_g - q)t_e - r_p t_p} t_e^{1.5} Sn(d_1)}{\sigma}$$

rho$_q$

$$\rho_{2,c} = \frac{\partial C}{\partial q} = -e^{(r_g - q)t_e - r_p t_p} t_e S n(d_1) \frac{\partial d_1}{\partial q}$$

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

$$\frac{\partial d_1}{\partial q} = \frac{-\sqrt{t_e}}{\sigma}$$

Combining the above equations gives:

$$\rho_{2,c} = \frac{e^{(r_g - q)t_e - r_p t_p} t_e S n(d_1)}{\sigma}$$

Theta

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

$$\frac{\partial d_1}{\partial \tau} = \frac{r_g - q}{\sigma\sqrt{t_e}} - \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{2\sigma t_e^{1.5}} + \frac{\sigma}{4\sqrt{t_e}}$$

Combining the equations above gives:

$$\Theta_c = e^{(r_g - q)t_e - r_p t_p}(-q + -r_p + r_g) S N(d_1) - e^{(r_g - q)t_e - r_p t_p} S n(d_1)\left(\frac{r_g - q}{\sigma\sqrt{t_e}} - \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{2\sigma t_e^{1.5}} + \frac{\sigma}{4\sqrt{t_e}}\right)$$

7.3.4 Asset or Nothing Put Options

NPV

$$P = Se^{(r_g - q)t_e - r_p t_p} N(-d_1)$$

$$d_1 = \frac{\ln\left[\dfrac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{\sigma\sqrt{t_e}} + \frac{\sigma\sqrt{t_e}}{2}$$

$$n(d_2) = \frac{e^{\frac{-d_2^2}{2}}}{\sqrt{2\pi}}$$

Delta

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

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

Combining the equations above gives:

$$\Delta_p = e^{(r_g - q)t_e - r_p t_p}\left(N(-d_1) - \frac{n(-d_1)}{\sigma\sqrt{t_e}}\right)$$

Delta Forward

$$\Delta_{p,\,forward} = \frac{\partial P}{\partial F} = e^{-r_p t_p}\left(Fn(-d_1)\frac{\partial - d_1}{\partial S} + N(-d_1)\right)$$

$$\frac{\partial - d_1}{\partial F} = \frac{\partial}{\partial S}\frac{\ln S - \ln K}{\sigma\sqrt{t_e}} = -\frac{1}{F\sigma\sqrt{t_e}}$$

Combining the equations above gives:

$$\Delta_{p,\,forward} = e^{-r_p t_p}\left(N(-d_1) - \frac{n(-d_1)}{\sigma\sqrt{t_e}}\right)$$

Gamma

$$\Gamma_p = \frac{\partial^2 P}{\partial S^2} = \frac{\partial}{\partial S} e^{(r_g - q)t_e - r_p t_p} \left(N(-d_1) - \frac{n(-d_1)}{\sigma\sqrt{t_e}} \right)$$

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

because

$$\frac{\partial - d_1}{\partial S} = \frac{\partial}{\partial S}\frac{(\ln S - \ln K)}{\sigma\sqrt{t_e}} = -\frac{1}{S\sigma\sqrt{t_e}}$$

and

$$\frac{\partial n(-d_1)}{\partial S} = \frac{n(-d_1) * -d_1}{S\sigma\sqrt{t_e}}$$

The above equations can be combined to give:

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

Vega

$$\nu_p = \frac{\partial P}{\partial \sigma} = e^{-r_p t_p} S n(-d_1) \frac{\partial - d_1}{\partial \sigma}$$

$$\frac{\partial d_1}{\partial \sigma} = -\frac{\ln\left[\dfrac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{\sigma^2\sqrt{t_e}} + \frac{\sqrt{t_e}}{2}$$

Combining the above equations gives:

$$\nu_c = -e^{(r_g - q)t_e - r_p t_p} S n(-d_1) \left(-\frac{\ln\left[\dfrac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{\sigma^2\sqrt{t_e}} + \frac{\sqrt{t_e}}{2} \right)$$

$\mathbf{rho_{r_p}}$

$$\rho_{p,p} = \frac{\partial P}{\partial r_p} = -t_p e^{(r_g - q)t_e - r_p t_p} S N(-d_1)$$

$\mathbf{rho_{r_g}}$

$$\rho_{g,p} = \frac{\partial P}{\partial r_g} = e^{-r_p t_p} S e^{(r_g - q)t_e} n(-d_1) \frac{\partial - d_1}{\partial r_g}$$

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

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

Combining equations gives:

$$\rho_{g,p} = -\frac{e^{(r_g - q)t_e - r_p t_p} Sn(d_1) t_e^{1.5}}{\sigma}$$

rho$_q$

$$\rho_{2,p} = \frac{\partial P}{\partial q} = e^{(r_g - q)t_e - r_p t_p} Sn(-d_1) \frac{\partial - d_1}{\partial q}$$

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

$$\frac{\partial - d_1}{\partial q} = \frac{\sqrt{t_e}}{\sigma}$$

Combining the above equations gives:

$$\rho_{2,p} = -\frac{e^{(r_g - q)t_e - r_p t_p} Sn(d_1) t_e^{1.5}}{\sigma}$$

Theta

Defining $t_1 = \tau - T_1$, $t_2 = \tau - T_2$, and $t_3 = \tau - T_3$:

$$\Theta_p = -\frac{\partial P}{\partial \tau} = e^{(r_g - q)t_e - r_p t_p}(q + r_p - r_g)SN(-d_1) - e^{(r_g - q)t_e - r_p t_p} Sn(-d_1)\frac{\partial - d_1}{\partial \tau}$$

$$\frac{\partial d_1}{\partial \tau} = \frac{r_g - q}{\sigma\sqrt{t_e}} - \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{2\sigma t_e^{1.5}} + \frac{\sigma}{4\sqrt{t_e}}$$

Combining the equations above gives :

$$\Theta_p = e^{(r_g - q)t_e - r_p t_p}(-q - r_p + r_g)SN(-d_1) - e^{(r_g - q)t_e - r_p t_p} Sn(-d_1)\left(-\frac{r_g - q}{\sigma\sqrt{t_e}} + \frac{\ln\left[\frac{Se^{-qt_e}}{Ke^{-r_g t_e}}\right]}{2\sigma t_e^{1.5}} - \frac{\sigma}{4\sqrt{t_e}} \right)$$

7.4 Barrier Options

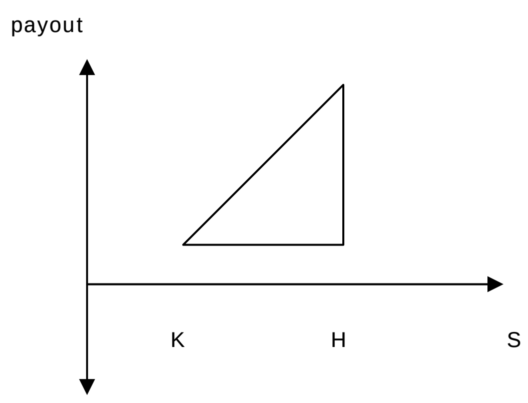
There are several types of Barrier Options that are implemented in the Calypso system and are discussed in the following section.

7.4.1 Simple Barrier Options

The Simple Barrier Option is a European style option whose payoff depends on the spot, strike and barrier levels. This particular option does not have a continuous barrier found in the Standard Barrier Option. The payout for a simple barrier call is described as follows:

- if S < K the payoff is 0
- if K < S < Barrier the payoff is S-K
- if S > Barrier the payoff is 0

and can be graphed as below:



Where K is the strike, H is the barrier, and S is the spot price.

Calculating the price and sensitivities of the Simple Barrier Option can be done by creating a portfolio of options which replicate the payoff of a Barrier Option. The portfolios for recreating simple barrier options are provided below.

Simple Barrier Call = Plain Vanilla Call Option $_{\text{with strike K}}$ – Asset or Nothing Binary Option $_{\text{with strike = barrier}}$

Simple Barrier Put = Plain Vanilla Put Option $_{\text{with strike K}}$ – Cash or Nothing Binary Option $_{\text{with strike = barrier}}$ – Plain Vanilla Put option $_{\text{with strike = barrier}}$

7.4.2 Standard Barrier Options

Standard Barrier Options have at least one continuous barrier. Each barrier in the Standard Barrier Option determines either when the option becomes live (knock-in) or when the option is ceases to exist (knock-out). A reverse barrier option is an option that has a barrier that is in the money. A knock-in barrier allows activates the option. If an option has a knock-in barrier, the spot price must cross this barrier in order for the option to be activated. If this barrier threshold is not met, then the option expires worthless at its pre-specified expiration date. There are both up and in and down and in options. The up and in option has a barrier that is higher than the spot price at issue and the down and in has a spot price that is higher than the barrier.

If the spot price crosses a knock-out barrier level, then the option ceases to exist and the payoff immediately becomes zero. A down and out barrier option ceases to exist when the spot price moves below the barrier and an up and out barrier option ceases to exist when the spot price moves above the barrier.

The equations for the standard barrier options are shown below.

for H > K :

$$\text{call}_{\text{up and in}} = Se^{-qt_3} N(x_1) - Ke^{-rt_2} N(x_1 - \sigma\sqrt{t_1}) - Se^{-qt_3}\left(\frac{H}{S}\right)^{2\lambda} (N(-y) - N(-y_1))$$

$$+ Ke^{-rt_2}\left(\frac{H}{S}\right)^{2\lambda-2} (N(-y + \sigma\sqrt{t_1}) - N(-y_1 + \sigma\sqrt{t_1}))$$

$$\text{call}_{\text{up and out}} = \text{call}_{\text{plain vanilla}} - \text{call}_{\text{up and in}}$$

$$\text{put}_{\text{down and out}} = 0$$

$$\text{put}_{\text{down and in}} = \text{put}_{\text{plain vanilla}}$$

For H ≥ K :

$$\text{call}_{\text{down and out}} = Se^{-qt_3} N(x_1) - Ke^{-rt_2} N(x_1 - \sigma\sqrt{t_1}) - Se^{-qt_3}\left(\frac{H}{S}\right)^{2\lambda} N(y_1)$$

$$+ Ke^{-rt_2}\left(\frac{H}{S}\right)^{2\lambda-2} N(y_1 - \sigma\sqrt{t_1})$$

$$\text{call}_{\text{down and in}} = \text{call}_{\text{plain vanilla}} - \text{call}_{\text{down and out}}$$

$$\text{put}_{\text{up and in}} = -Se^{-qt_3}\left(\frac{H}{S}\right)^{2\lambda} N(-y) + Ke^{-rt_2}\left(\frac{H}{S}\right)^{2\lambda-2} N(-y - \sigma\sqrt{t_1})$$

$$\text{put}_{\text{up and out}} = \text{put}_{\text{plain vanilla}} - \text{put}_{\text{up and in}}$$

for H < K :

$$\text{put}_{\text{down and in}} = -Se^{-qt_3} N(-x_1) + Ke^{-rt_2} N(-x_1 + \sigma\sqrt{t_1}) + Se^{-qt_3}\left(\frac{H}{S}\right)^{2\lambda} (N(y) - N(y_1))$$

$$- Ke^{-rt_2}\left(\frac{H}{S}\right)^{2\lambda-2} (N(y - \sigma\sqrt{t_1}) - N(y_1 - \sigma\sqrt{t_1}))$$

$$\text{put}_{\text{down and out}} = \text{put}_{\text{plain vanilla}} - \text{put}_{\text{down and in}}$$

for H ≤ K :

$$\text{call}_{\text{down and in}} = Se^{-qt_3}\left(\frac{H}{S}\right)^{2\lambda} N(y) - Ke^{-rt_2}\left(\frac{H}{S}\right)^{2\lambda-2} N(y - \sigma\sqrt{t_1})$$

$$\text{call}_{\text{down and out}} = \text{call}_{\text{plain vanilla}} - \text{call}_{\text{down and in}}$$

$$\text{call}_{\text{up and out}} = 0$$

$$\text{call}_{\text{up and in}} = \text{call}_{\text{plain vanilla}}$$

$$\text{put}_{\text{up and out}} = -Se^{-qt_3} N(-x_1) + Ke^{-rt_2} N(-x_1 + \sigma\sqrt{t_1}) + Se^{-qt_3}\left(\frac{H}{S}\right)^{2\lambda} N(-y_1) - Ke^{-rt_2}\left(\frac{H}{S}\right)^{2\lambda-2} N(-y_1 + \sigma\sqrt{t_1})$$

$$\text{put}_{\text{up and in}} = \text{put}_{\text{plain vanilla}} - \text{put}_{\text{up and out}}$$

where :

$$\lambda = \frac{r - q + \frac{\sigma^2}{2}}{\sigma^2}$$

$$y = \frac{\ln(\frac{H^2}{SX})}{\sigma\sqrt{t_1}} + \lambda\sigma\sqrt{t_1}$$

$$y_1 = \frac{\ln(\frac{H}{S})}{\sigma\sqrt{t_1}} + \lambda\sigma\sqrt{t_1}$$

$$x_1 = \frac{\ln(\frac{S}{H})}{\sigma\sqrt{t_1}} + \lambda\sigma\sqrt{t_1}$$

7.4.3 Barrier-At-Expiry Options

A Barrier-At-Expiry Option has a barrier only on the expiry date. This is sometimes referred to as a *European barrier*, as opposed to a continuous *American barrier*.

A Knock-Out At-Expiry Option will pay out as a vanilla option if the price on the barrier expiry date is not past the barrier. A Knock-In At-Expiry Option will pay out as a vanilla option if the price is past the barrier on the barrier expiry date.

Because the barrier is only at expiry, "Down and In" is the same as "Up and Out", etc. With spot price S, strike K, and barrier H, the payouts can be defined with the help of the step function

Theta(x) = 1 if x >= 0, 0 otherwise.

Then the payouts are:

Up and In Call = Down and Out Call

Max(S-K, 0) * Theta(S - T)

Down and In Call = Up and Out Call

Max(S - K, 0) * Theta (T - S)

Up and In Put = Down and Out Put

Max(K - S, 0) * Theta(S - T)

Down and In Put = Up and Out Put

Max(K - S, 0) * Theta (T - S)

Barrier At-Expiry Options can be valued by replication using vanilla puts, calls, and simple cash-or-nothing digitals. The following table shows the valuation for all cases. For illustration the currency pair is USD/JPY, with JPY the quoting currency.

Put/Call	***Spot(S) vs Barrier(H)***	***Strike(K) vs Barrier(H)***	***Replication***
Yen Call	KI(H>S)	K<=H	0
	KI(H>S)	K>H	YenCall(K)-YenCall(H)-(K-H)*DigYenCall(H)
	KO(H>S)	K<=H	YenCall(K)
	KO(H>S)	K>H	YenCall(H)+(K-H)*DigYenCall(H)
	Rv KI (H<S)	K<=H	YenCall(K)
	Rv KI (H<S)	K>H	YenCall(H)+(K-H)*DigYenCall(H)
	Rv KO(H<S)	K<=H	0
	Rv KO(H<S)	K>H	YenCall(K)-YenCall(H)-(K-H)*DigYenCall(H)
Yen Put	KI(H>S)	K<=H	YenPut(H)+(H-K)*DigYenPut(H)
	KI(H>S)	K>H	YenPut(K)
	KO(H>S)	K<=H	YenPut(K)-YenPut(H)-(H-K)*DigYenPut(H)
	KO(H>S)	K>H	0
	Rv KI (H<S)	K<=H	YenPut(K)-YenPut(H)-(H-K)*DigYenPut(H)
	Rv KI (H<S)	K>H	0

Put/Call	Spot(S) vs Barrier(H)	Strike(K) vs Barrier(H)	Replication
	Rv KO(H<S)	K<=H	YenPut(H)+(H-K)*DigYenPut(H)
	Rv KO(H<S)	K>H	YenPut(K)

7.4.4 Double Barrier-At-Expiry Options

Like the single Barrier-At-Expiry options, Double Barrier-At-Expiry Options pay out as a put or a call depending on where the spot rate is with respect to the barriers on the expiry date. They are valued using replication with puts, calls, and simple digitals. The replication instruments are defined as follows. For illustration the currency pair is USD/JPY, with JPY the quoting currency. In this table, one always has A < B.

Rep. Instrument	Description	Replication
DigitalYenPut(A)	Digital Yen Put Option with strike A	
DigitalYenCall(A)	Digital Yen Call Option with strike A	
YenCall(A)	Yen Vanilla Call Option with strike A	
YenPut(A)	Yen Vanilla Put Option with strike A	
DIG(A, B)	Pays one unit of the quote currency (Yen) if and only if the FX rate falls between A and B at the expiry. (A<B)	DigitalYenPut(A) - DigitalYenPut(B)
PartialYenCall(A, B)	Yen Call with strike B but truncated at Spot = A. (A < B)	YenCall(B)-YenCall(A)-(B-A)*DigitalYenCall(A)
PartialYenPut(A, B)	Yen Put with strike A but truncated at Spot = B. (A < B)	YenPut(A)-YenPut(B)-(B-A)*DigitalYenPut(B)

The following tables below shows the replication of Double Barrier-At-Expiry Options for all cases.

Put/Call	Area where Option is alive	Strike(K) vs. Trigger(T1, T2, T1<T2)	Replication
Yen Call	Inside the Barriers	K< T1< T2	0
		T1< K< T2	PartialYenCall(T1, K)
		T1 <T2< K	PartialYenCall(T1, T2) + (K-T2)*DIG(T1, T2)
	Outside the Barriers	K< T1< T2	YenCall(K)
		T1< K< T2	YenCall(T1)+(T1-K)*DigitalYenCall(T1)
		T1 <T2< K	PartialYenCall(T2, K)+YenCall(T2)+(K-T2)*DigitalYenCall(T2)
Yen Put	Inside the Barriers	K< T1< T2	PartialYenPut(T1,T2)+(T1-K)*DIG(T1,T2)
		T1< K< T2	PartialYenPut(K,T2)
		T1 <T2< K	0
	Outside the Barriers	K< T1< T2	PartialYenPut(K,T1)+YenPut(T2)+(T2-K)*DigitalYenPut(T2)
		T1< K< T2	YenPut(T2)+(T2-K)*DigitalYenPut(T2)
		T1 <T2< K	YenPut(K)

7.4.5 Digital Single - Barrier Options

Digital Barrier Options are trigger-type Cash-or-Nothing Options whose payoff occurs if the barrier is touched during the lifetime of the option. The cash payoff can occur either when the barrier is breached or at the maturity of the option. The formulas for these types of options can be found in Haug, pp. 92-95.

7.4.6 Digital Double - Barrier Options

The cash payoff can occur either when the barrier is breached or at the maturity of the option. There are both knock-in and knock-out options. Currently, the Wystup approximation for double-no-touch options and the Pelsser approximation for double-touch options with instant payoff are implemented.

7.4.7 Adjustment for Discrete Barriers

Many options contracts specify the observations as daily closing prices. If this is the case, a correction can be made to the barrier level. Calypso implements the Broadie, Glasserman and Kou (1995) method, also found in Haug, p. 85. The barrier levels are adjusted by exp(0.5826 * volatility * sqrt(observations per year). The upper barrier is multiplied by this factor and the lower divided by the factor. The user can enter the number of observations per year for both Standard and Digital Barrier Options.

7.5 Asian Options

Arithmetic Average Strike Asian Options can be priced in Calypso. Average Spot options are priced using the Turnbull Wakeman approximation. The pricing equation for arithmetic average spot Asian Options can be seen below.

$$C = Se^{(b_a - r)t_e} N(d_1) - Ke^{-rt_e} N(d_2)$$

Details can be found in Haug, pp. 97 – 100.

7.6 LookBack Options

7.6.1 Floating Strike Lookback Options

Floating strike lookback options are options which pay either the maximum of the asset price at maturity minus the lowest price during the life of the option in the case of call options or the difference between the highest price during the option’s life and the spot price at maturity for put options. The Goldman, Sosin, and Gatto formula can be found in Haug, pp. 61-62 and is shown below.

$$C = Se^{(b-r)t_e} N(a_1) - S_{\min} e^{-rt_e} N(a_2)$$

$$+ Se^{-rt_e} \frac{\sigma^2}{2b} \left[\left(\frac{S}{S_{\min}} \right)^{-\frac{2b}{\sigma^2}} N\left(-a_1 + \frac{2b}{\sigma}\sqrt{t_e}\right) - e^{bt} N(-a_1) \right]$$

7.6.2 Fixed Strike Lookback Options

Fixed strike lookback options are options which pay either the maximum of the asset price at maturity minus the strike in the case of call options or the difference between the strike and the minimum price observed during the option's life. The formula can be found in Haug, pp. 63-64 and is shown below.

$$C = Se^{(b-r)t_e} N(d_1) - S_{\min} e^{-rt_e} N(d_2)$$

$$+ Se^{-rt_e} \frac{\sigma^2}{2b} \left[-\left(\frac{S}{X} \right)^{-\frac{2b}{\sigma^2}} N(d_1 - \frac{2b}{\sigma}\sqrt{t}) + e^{bt} N(d_1) \right]$$

Section 8. The Vanna-Volga Method

8.1 The Cost of Vega Hedging

The failure of the Black-Scholes model is most apparent in the pricing of barrier options. If there is a skew or smile in the volatility surface, which volatility should one use: at the strike, at the barrier level (and if there are two barriers, which one?), or something in between?

Furthermore, the fact that volatility is rarely constant produces the possibility of arbitrage trades. The most notorious case is the Reverse Knock-Out Barrier. Using standard Black-Scholes pricing, the barrier price has the following sensitivity to changes in volatility:

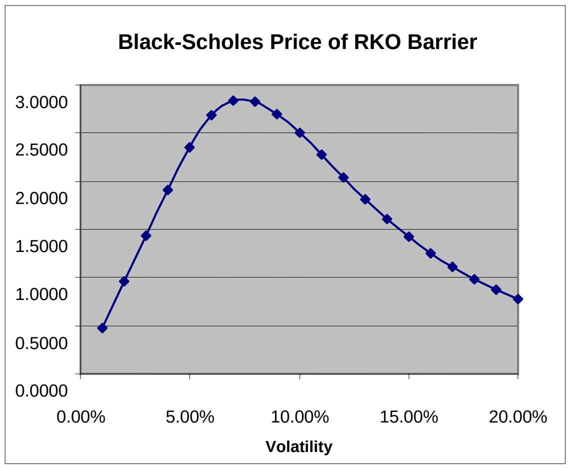


Fig 1. Knockout barrier option with expiry 1Y, Spot = 120, Strike = 120, Barrier = 140

A trader who purchases this option near the peak price will lose money if volatility moves in either direction. To avoid this it is necessary to vega-hedge the option. The hedge can be calculated using the same Black-Scholes model at the current volatility. To hedge volatility as well as the sensitivity of volatility to market changes will require vanilla options at more than one strike, but each strike is typically priced in the marketplace at a different volatility -- so there will be some difference from the single volatility used to compute the hedge. So there is an uncertainty in the pricing of the hedge (or in the accuracy of the hedge), which should be folded into the price of the barrier in order to protect the trader.

The need to vega-hedge barrier options is described by Webb (see the references). A method of taking account the uncertainty of the vega hedge for general exotic options was described by Hagan. In the FX market, the adjustment to hedge vega as well as its sensitivities was termed by Wystup the "Trader's Rule of Thumb" and is now more commonly known as the "Vanna Volga Method." The method computes an adjustment to the Black-Scholes price of an option to take into account the volatility smile or skew. The mathematical logic behind the method was developed more fully by Castagna and Mercurio (2007), and it is their approach that Calypso has implemented.

The following section describes the Vanna Volga concept following Castagna and Mercurio. Afterwards, the use of the method in Calypso is described.

References

A. Castagna and F. Mercurio, The Vanna-Volga Method for Implied Volatilities, Risk (Jan 2007)

Patrick Hagan, Adjusters: Turning Good Prices into Great Prices, Wilmott Magazine (November 2002)

A. Lipton and W. McGhee, Universal Barriers, Risk (May 2002)

Andrew Webb, The Sensitivity of Vega, Derivatives Strategy (Nov 1999)

Uwe Wystup, The Market Price of One-Touch options in Foreign Exchange Markets, Derivatives Week 12 14 (2003)

Uwe Wystup, Vanna-Volga Pricing, Frankfurt School of Finance and Management (July 2008)

8.2 The Adjustment Formula

Suppose one has a set of market prices C_i^{MKT} for vanilla calls with strikes K_i for some time to expiry, and one wishes to use this information to price another option V. Pricing these options under Black-Scholes with some chosen flat volatility σ^{BS} would produce values V_{BS} for the target option and C_i^{BS} for the vanilla calls. Under this model, using three vanilla calls a portfolio can be formed out that is hedged with respect to volatility sensitivities vega, vanna (DVega/DSpot, the sensitivity of vega to spot) and volga (DVega/DVol, the sensitivity of vega to volatility). This will have value

$$P_{BS} = V_{BS} + \sum_i x_i C_i^{BS}$$

where x_i is the number of calls at strike K_i, i = 1,2,3. The x_i can be solved for by requiring the portfolio have zero vega, vanna and volga.

But this model does not produce the correct option prices even for the vanilla calls. Suppose instead there is a "true" but unknown model that does produce the correct vanilla call prices. Under the "true" model the portfolio has value

$$P_{TRUE} = V_{TRUE} + \sum_i x_i C_i^{MKT}$$

At expiry the portfolio necessarily has the same value under either model. Castagna and Mercurio argue that, because of the hedging, the portfolios have the same value at all prior times, also.

$$V_{TRUE} + \sum_i x_i C_i^{MKT} = V_{BS} + \sum_i x_i C_i^{BS}$$

Which can be written

$$V_{TRUE} = V_{BS} + \sum_i x_i (C_i^{BS} - C_i^{MKT})$$

It will be noted that, regardless of the details of the argument, one would expect a linear correction for the "true" option value to be expressed in this form, as it uses all of the information given in the market, that is, the differences between market and theoretical prices of vanilla calls. This can be considered as a Taylor expansion around a flat volatility surface. Given this form, by further requiring that P_{BS} have zero vega, vanna, and volga, it follows that P_{TRUE} formed out of market priced calls also has zero vega, vanna and volga, and one will have a volatility-hedged portfolio -- provided one can neglect any dependence of the x_i on volatility and spot. So if one selects a volatility σ^{BS} so that the theoretical prices are near the market prices of calls, the correction term will be small, and the linear form for the correction would be a reasonable approximation.

Because of the choice of weights, the sensitivity to the volatility σ^{BS} has been eliminated, leaving only a sensitivity to the market call options at the market prices. The remaining volatility risk of the adjusted price is the last term,

$$-\sum_i x_i C_i^{MKT}$$

which can be hedged by buying or selling the vanilla calls at market prices. As Hagan remarks, the "adjuster" has moved the vega risk from model-priced options to market-priced options.

The equation for the adjusted price can be written in a more suggestive form. When solved for, the weights x_i will be combinations of the Black-Scholes vega, vanna and volga of the option V and the vanilla calls. Therefore, substituting in the values of x_i the correction term can be rewritten in terms of the V sensitivities,

$$V_{TRUE} = V_{BS} + y_{VEGA} * Vega(V_{BS}) + y_{VANNA} * Vanna(V_{BS}) + y_{VOLGA} * Volga(V_{BS})$$

The *y* coefficients depend only on the vanilla call valuations and prices, not on the target option V. They are calculated using

$$y_a = (C_i^{BS} - C_i^{MKT})(\mathcal{C}^{-1})_{ia})$$

In this formula, the *'a'* subscript stands for either Vega, Vanna, or Volga, and the matrix symbol c_{ia} is the *'a'* sensitivity of the *ith* vanilla option.

For the derivation of this formula please see the Calypso white paper, "The Vanna-Volga Method of Option Pricing in Calypso."

8.3 Barrier Option Treatment

One more change to the formula is commonly made when pricing barrier options, arising from the fact that a knock-out barrier option, for example, would no longer need to be hedged once it is knocked out. The result is to reduce the cost of the vega hedge:

$$V_{\text{VannaVolga}} = V_{BS} + p\sum_i x_i (C_i^{BS} - C_i^{MKT})$$

The factor '*p*' is the adjustment, which is typically the probability the barrier is not touched (so the full Market versus BS difference is realized at expiry). This is not well justified mathematically as of yet, and there are a variety of opinions as to the best '*p*' to use for each option type, including taking into account the proximity of the barrier(s). The best practice may be to calibrate 'p' to actual market quotes of barrier options.

8.4 Volatility Interpolation

By applying the Vanna Volga formula to vanilla options, a volatility surface can be defined through interpolation and extrapolation of the market implied volatilities. For a call option at any strike K, one finds the adjusted price

$$C_{K,\text{VannaVolga}} = C_K(\sigma^{BS}) + \sum_i x_i (C_i^{BS} - C_i^{MKT})$$

where σ^{BS} can be the ATM volatility or the implied volatility of one of the market-priced calls. Then finding the Black-Scholes implied volatility of the adjusted price produces an interpolated or extrapolated volatility. Castagna and Mercurio (2007) provide an approximation to this formula that provides a smooth and fast interpolation method.

8.5 Comparison with Wystup

Wystup's form of the Vanna Volga method is expressed in terms of market risk reversal (RR) and butterfly (BF) prices, and can be viewed as a faster approximation to the formulas given above. Wystup's formula is

$$V_{WystupVannaVolga} = V_{BS} + p * y_{VANNA} * Vanna(V_{BS}) + p * y_{VOLGA} * Volga(V_{BS})$$
$$y_{VANNA} = (RR^{MKT} - RR^{BS}) / Vanna(RR^{BS})$$
$$y_{VOLGA} = (BF^{MKT} - BF^{BS}) / Volga(BF^{BS})$$

The factor *p* is again the correction for barrier option, and can vary with option type. One notices the similarity between this and the expressoin for y_a given above. In Wystup's approximation, the pure Vega coefficient is zero.

8.6 Calypso Barrier Option Pricer

Calypso provides the Vanna Volga adjustment for barrier options as a standard feature of the pricer FXOptionBarrier. This pricer provides the standard Black-Scholes valuation of barriers and will add in the Vanna Volga adjustment if the user chooses. This is done with the following pricing parameter:

> **USE_VANNA_VOLGA_ADJ** If true, adds the Vanna Volga adjustment to the Black-Scholes price calculation. Sensitivities (delta, gamma, etc.) will include affects from the adjustment. If false, the standard Black-Scholes calculation is performed. The pricer measure **TV**, Theoretical Value, will show the standard Black-Scholes NPV in either case.

This can be set either for one trade in a trade window or for all trades using the Pricing Environment Parameters or the Pricer Config Model Parameters tab.

For volatility input, the pricer just employs the usual volatility surface associated with the trade in order to derive smile information. This does not need to have Risk Reversal or other quotes; it can be a manually-defined or externally-loaded surface. The pricer will interpolate the necessary smile information. If no volatility surface is provided, no adjustment will be calculated.

At present only barrier options have the Vanna Volga adjustment.

8.7 Calculation Choices

The Vanna Volga method is not completely mathematically defined, particularly for barrier options. The current implementation of Calypso makes the following choices.

Matrix Calculations. Calypso performs the full matrix calculation of the Vanna Volga weights as described by Castagna and Mercurio, and full calculations of Black-Scholes vega, vanna, and volga. It does not use the approximations of Wystup. In testing, we find the two methods usually give similar results.

Formula Inputs.

- Volatilities are derived from the volatility surface assigned in the Pricer Config. There is currently no separate ability to specify RR or FLY quotes as parameters in the trading window.
- σ^{BS} is chosen to be the volatility of an at-the-money forward (volatility at strike of a delta neutral straddle). This is derived from the volatility surface.
- The three vanilla option inputs are calls with strikes corresponding to the at-the-money forward and the 25 and 75 forward delta without premium.
- The factor "p" is the value of a No Touch barrier option (double or single, as appropriate) of unit notional. Wystup has recommended varying "p" for the option type, but this has not been implemented into Calypso pending further research. Castagna and Mercurio note that there is no mathematical justification yet for the choice of "p" or the best methodology to apply Vanna Volga to barriers.

Log Category. When using the barrier price, details of the adjustment calculation can be viewed through use of the Log category "VannaVolga".

Bid-ask spreads. Vanna Volga is not considered a reliable way to produce bid-ask spreads. Calypso advises computing option values with MID quotes, and subsequently apply spreads observed in the marketplace.

8.8 Volatility Interpolation (Future Enhancement)

In principle, Vanna Volga can be used to interpolate and extrapolate market volatilities. At present Calypso has not made this available to users. Calypso does not restrict users to three points on a surface, and the application of the method to arbitrary surfaces is the subject of further investigation. Clients are encouraged to enquire if interested.

8.9 Example

The following example shows the calculation of the Vanna Volga adjustment for a 3-month Reverse Knock-Out barrier option.

Market data

Spot	96.68
USD MM %	0.74
JPY MM %	0.33

ATMF	25 Delta RR	25 Delta Fly
14.95	-3.1	0.35

Barrier option trade

CALL	USD
PUT	JPY

Strike	96.50
Barrier	103.00
Trade Date	4-Jun-09
Spot Date	8-Jun-09
Expiry	7-Sep-09
Delivery	9-Sep-09

Calibration options

From the volatility surface, find the call option strikes corresponding to 25 and 75 forward delta, and 0 forward delta straddle (the strike where call forward delta = put forward delta). Compute the ATM price, vega, vanna and volga, and compute the market price using the actual volatility quote from the surface.

All options have:

Time to Expiry	0.260274
Time Spot to Payment	0.254795

From the market data given above, the Calypso calculation finds the following call options.

Strike	101.4133	96.8341	91.5142
Volatility	0.135312	0.147121	0.165818
Mkt fwd delta	0.25039	0.50086	0.75139
ATMPrice	1.151413	2.767215	6.038102
ATMVega	16.27456	19.64015	14.76627
ATMVanna	1.543483	0.196	-1.38422
ATMVolga	46.67879	-0.02383	51.56702
MktPrice	0.962723	2.767215	6.322342

Barrier option Black-Scholes values

For the barrier option, the standard Calypso pricer produces the following Black-Scholes results using the ATM volatility.

ATM Price (TV)	0.340356243
ATM Vega	-5.003570891
ATM Vanna	-0.152221609
ATM Volga	77.88258927

Calculation of weights

From the information calculated for the calibration options, form the 3X3 matrix of vega, vanna and volga for the vanilla calls. Invert the matrix and multiply by the differences between the market and ATM price of the calls. This produces the y_a weights, as described in Section 2.4. The following results are calculated by Calypso:

Vega weight	0.001613199
Vanna weight	-0.16115823
Volga weight	0.000724

No Touch probability

To take into account the possibility that the barrier will knock-out, eliminating the risk of volatility hedging, the adjustment will be multiplied by the probability of not touching the barrier. This is the price of a No Touch option of the same characteristics as the barrier option being priced. For this, the ATM volatility is used. This gives

p	0.617780041

Calculation of adjusted barrier price

Now each weight can be multiplied by the barrier option ATM sensitivities and added to the ATM price. The formula is

$$V_{Adjusted} = V_{BS} + p * [y_{VEGA} * Vega(V_{BS}) + y_{VANNA} * Vanna(V_{BS}) + y_{VOLGA} * Volga(V_{BS})]$$

Inserting the values calculated above, we find

$$V_{Adjusted} = 0.340356243 + 0.617780041 * [0.001613199 * (-5.003570891)$$
$$+ (-0.16115823) * (-0.152221609)$$
$$+ 0.000724 * 77.88258927]$$

Resulting in

$$V_{Adjusted} = 0.385364917$$

which is the adjusted price of the barrier in terms of JPY per unit USD notional.

Without adjustment by the probability *p* one would have a value of about 0.413. One can see the choice of probability, which is somewhat ad hoc, has a significant effect on the result. This is one reason the Vanna Volga method should be regarded as at best a guide, not an absolute result. It may be that calibrating *p* to actual market quotes of barrier options would improve the method, which is a potential future enhancement in Calypso.

Comparison with Wystup

To find values for the Risk Reversal we subtract the 75-delta call (which prices the same as a 25-delta put) from the 25-delta call. To find values for the Butterfly we sum them and subtract twice the ATM values. Using the calibration option information found previously, one computes the following.

ATM RR Price = 1.151413 - 6.038102 = -4.886689

ATM RR Vanna = 1.543483 - (-1.38422) = 2.927703

Mkt RR Price = 0.962723 - 6.322342 = -5.359619

ATM Fly Price = (1.151413 + 6.038102) - 2 * = 1.655085

ATM Fly Volga = (46.67879 + 51.56702) - 2 * (-0.02383) = 98.29347

Mkt Fly Price = (0.962723 + 6.322342) - 2 * 2.767215 = 1.750635

Using Wystup's formulas, one finds:

Vanna weight = (-5.359619 - (-4.886689)) / 2.927703 = -0.161536194

Volga weight = (1.750635 - 1.655085) / 98.29347 = 0.000972089

Comparison with the previously computed table using matrix inversion shows very similar values, apart from the dropping of the vega term. One finds the adjusted price

$V_{Adjusted}$ = 0.340356243 + 0.617780041 * (-0.161536194*(-0.152221609) + 0.000972089 * 77.88258927)
= 0.402318412

Without the p coefficient, the price would be about 0.441.

Section 9. The Heston Model

If the Black-Scholes model were an adequate explanation of option prices, all options at the same expiry on a single asset would be quoted at the single expected volatility of that asset. But in the markets, options at different strikes often have different implied volatilities. This is typically attributed to the fact that the asset's volatility is not known and constant but unknown and ever-changing.

The Heston model is one of the most popular models that takes into account uncertain volatility. It directly describes some commonly observed features of asset volatility. The following charts demonstrate these features. The top chart is the S&P 500 over the course of the year 2008, while the bottom chart is the implied volatility of S&P 500 index options over the same period. Volatility obviously is not constant over the period, but there are also some other apparent features.

Top: S&P 500, Jan 2008-Jan 2009.

Bottom: VIX implied volatility index, same period.

First, volatility is negatively correlated to asset price: lower prices mean higher volatility. And second, volatility tends to oscillate around a fairly-constant mean level, although there can be an abrupt change in that mean level at times of market disruption.

The Heston model incorporates this behavior of volatility in the following parameters:

- **Risk-free rate**
- **Current lognormal volatility**
- **Volatility of volatility, the uncertainty in the volatility**
- **Mean volatility, the long-term average volatility**
- **Mean reversion speed, the speed at which volatility will return to the mean level once it deviates from it**
- **Correlation of volatility and asset price, which is typically negative.**

Choosing values for these parameters completely specifies the Heston model, which can then be employed to price options. The parameters are chosen to fit market prices of vanilla puts and calls, and one reason for the popularity of the model is the relative ease with which this can be done compared with competing models. Furthermore, by setting correlation, volatility of volatility and reversion speed to zero, the Heston model reduces to the Black-Scholes model.

Barrier options are a common option type that are not well priced in the single-volatility Black-Scholes model. The following sections describe how to use the Heston model to price barrier options in Calypso.

References

The smile problem

- Lipton, “The Volatility Smile Problem”, *Risk* (Feb 2002)
- Lipton & McGhee, “Universal Barriers”, *Risk* (May 2002)
- Jim Gatheral, *The Volatility Surface* (Wiley, 2006)

The Heston model

- Heston, S. L., "A closed-form solution for options with stochastic volatility with applications to bond and currency options", Review of Financial Studies 6 (1993) 327-343
- Lipton, Alexander, *Mathematical Methods for Foreign Exchange*, (World Scientific 2001), p. 233 ff., 375 ff.
- Gatheral, op. cit.
- Andersen, Leif, "Simple and efficient simulation of the Heston stochastic volatility model", *Journal of Computational Finance* 11 3 (Spring 2008)
- Kahl, C. and P. Jaeckel, "Not-so-complex logarithms in the Heston Model", *Wilmott Magazine* (Sep 2005)

Calibration

Price, Kenneth and Rainer M. Storm, *Differential Evolution: A Practical Approach to Global Optimization* (Springer, 2005)

9.1 Product Coverage

Release 11 prices standard single and double barrier puts and calls.

9.2 Pricers

For FX options, there are two implementation of Heston pricing of barriers

- **PricerFXOptionBarrierHestonMC** - Monte Carlo implementation
- **PricerFXOptionBarrierHestonFD** - Finite Difference implementation

The Monte Carlo implementation employs Anderson's QE algorithm, currently considered the fastest and most reliable algorithm. Accurate simulation of the Heston model requires many steps per path; Calypso typically uses a simulation frequency of three points per day for medium-term maturities. Run times on a 1.7 GHz processor are tested to be 0.0009 milliseconds per simulation point. The number of Monte Carlo paths is controlled by the parameter **HESTON_NUM_MC_PATHS**.

The Finite Difference pricer employs an ADI operator-splitting algorithm to solve the two-dimensional Heston partial differential equation. In principle there are several ways to perform the splitting and to preserve boundary conditions; after numerous experiments, Calypso has selected a method it has seen to be most stable and reliable, related to that of Lipton. The grid size is determined dynamically depending on the location of the barriers.

The Finite Difference pricer is much faster than the Monte Carlo pricer, especially when it comes to computing standard sensitivities: these are all available at one finite difference evaluation, while each Monte Carlo sensitivity employs a new evaluation.

9.3 Pricing Parameters

The Heston model has four more input parameters than the Black-Scholes model. In a trade window for an FX Barrier option, these can be set manually as Pricing Parameters.

- **VOLATILITY** : The initial volatility for the Heston process. (In the FX Option trade window, this is a field rather than a pricing parameter.)
- **HESTON_DRIFT** : The risk-free rate of the underlying asset price; if not set explicitly, the rate will be read from the discount curve as of the option maturity.
- **HESTON_MEAN_VAR** : Mean variance (volatility squared), the long-term average variance
- **HESTON_VAR_REV_SPEED** : Mean reversion speed, the speed at which variance will return to the mean level once it deviates from it

- **HESTON_VOL_VOL** : Volatility of volatility, the uncertainty in the volatility
- **HESTON_CORR** : Correlation of volatility and asset price; this is typically negative.

All quantities are entered as decimals, not as percents. The following table gives an example of parameter values as they would be manually entered. Please note when volatility is used and when variance (volatility squared) is used, which attempts to bridge to types of terminology: in academic literature the Heston model is usually described in terms of variance, but in practical trading one more often refers to volatility. The last column shows the symbols often used for each pricing parameter in the literature, but as different authors use different symbols it is least ambiguous to use the actual parameter name.

Pricing Parameter	Example Value	Comment	Symbols
VOLATILITY	0.20	20% initial volatility	$V_0, \sqrt{V_0}$
HESTON_DRIFT	0.03	3% asset drift	μ
HESTON_MEAN_VAR	0.04	Square of a 20% mean volatility	θ
HESTON_VAR_REV_SPEED	0.5	Reversion coefficient of .5	κ
HESTON_VOL_VOL	1.0	Diffusive term coefficient of 0.9 in volatility process	$\sigma, \Sigma, \varepsilon, \xi$
HESTON_CORR	-0.9	Asset price to volatility correlation of -0.9	ρ

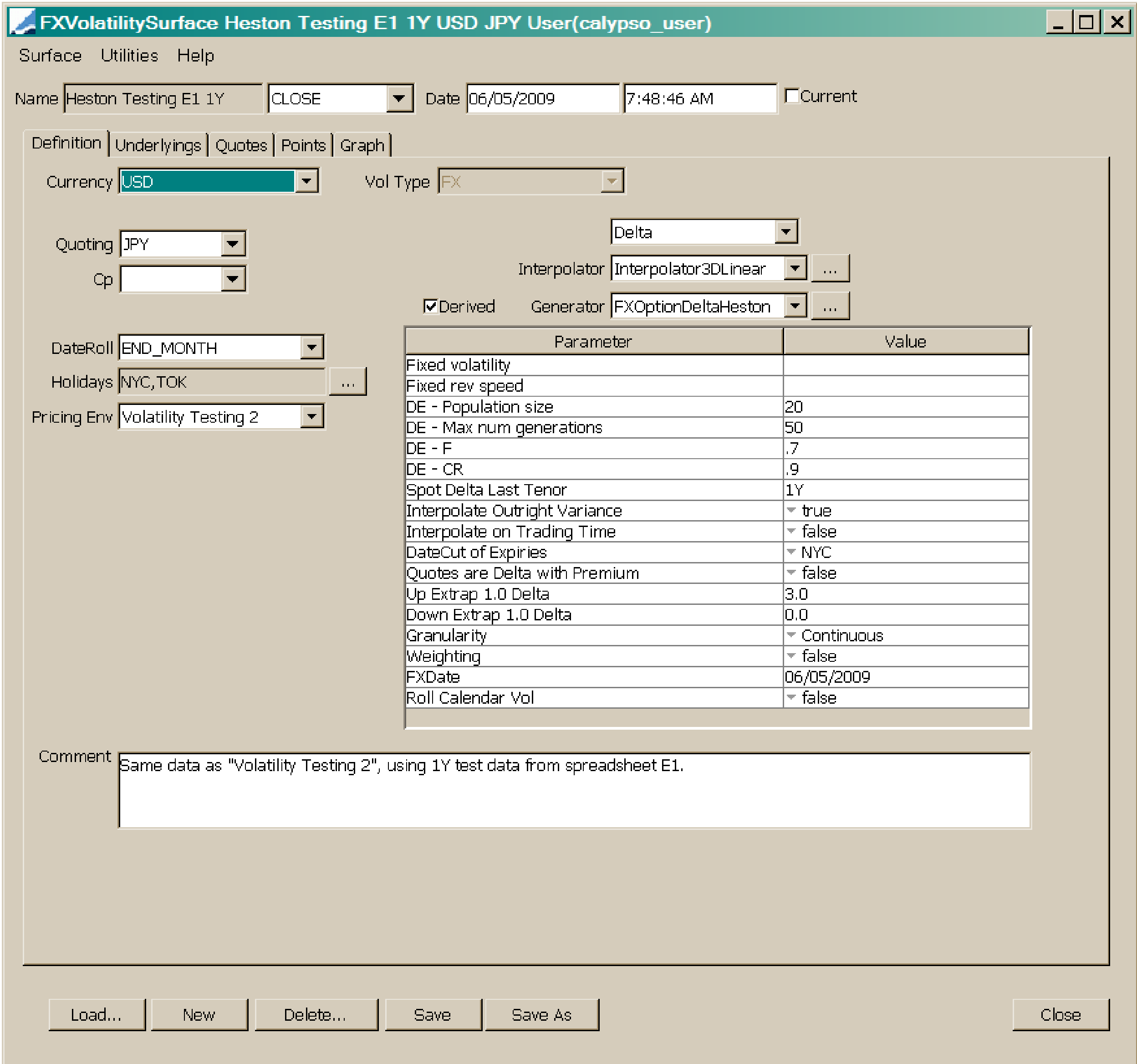
9.4 Model Calibration

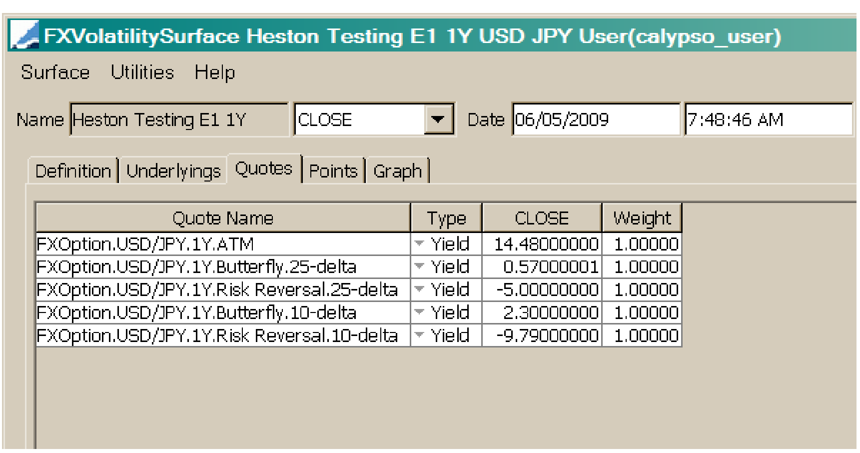
Calypso provides a means to calibrate the Heston model to a set of vanilla puts and calls. The calibrated values can then be used in as the Pricing Parameters for the barrier pricers discussed in the previous section. The calibration is performed using a Differential Evolution genetic algorithm that has shown itself to be faster and more reliable than alternative mechanisms.

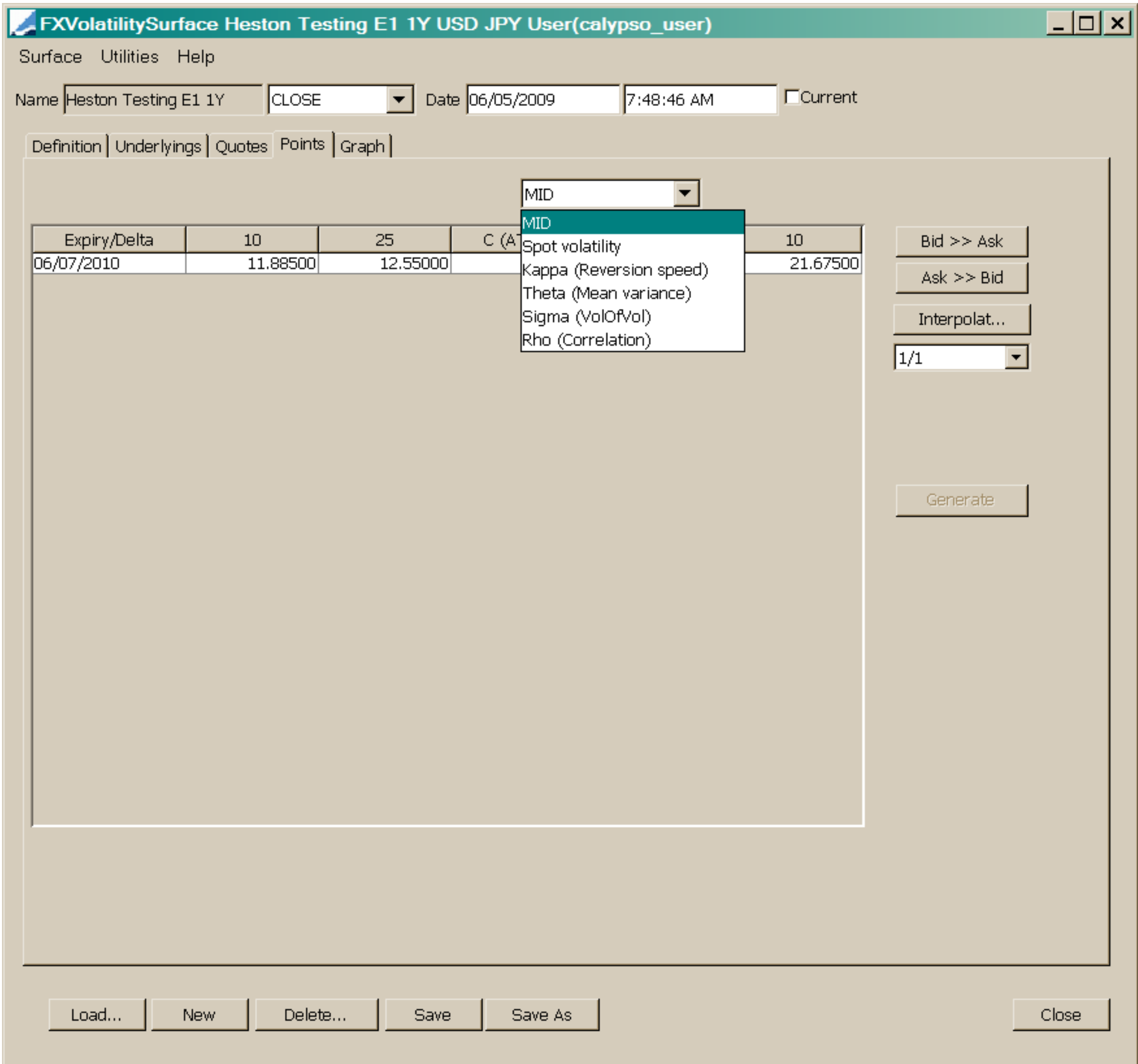
9.4.1 Surface Generation with FXOptionDeltaHeston

To perform a calibration on FX options, create an FX volatility surface using the generator **FXOptionDeltaHeston**. This performs both the usual surface generation of FXOptionDelta and the calibration of the Heston model. Other than the choice of generator, the procedure is the same as any other volatility surface: select underlying instruments, input quotes, set generator parameters, and then generate the points.

The output of the Heston calibration is seen using the Points selection menu on the Points tab. Usually this menu will display "MID", "LAST", etc.; for the Heston generator it displays the list of parameters corresponding to the pricing parameters show previously. Selecting a variable name will display the calibrated value. See the following screenshots for an example. There are five variables listed: the Asset Drift is not calibrated, but is instead read from the discount curves of the currency pair.


FXVolatilitySurface Heston Testing E1 1Y USD JPY User(calypso_user)
Surface Utilities Help
Name Heston Testing E1 1Y CLOSE Date 06/05/2009 7:48:46 AM Current
Definition Underlyings Quotes Points Graph
Currency USD
Vol Type FX
Quoting JPY
Cp
Delta
Interpolator Interpolator3DLinear ...
Derived
Generator FXOptionDeltaHeston ...
DateRoll END_MONTH
Holidays NYC,TOK ...
Pricing Env Volatility Testing 2
Parameter Value
Fixed volatility
Fixed rev speed
DE - Population size 20
DE - Max num generations 50
DE - F .7
DE - CR .9
Spot Delta Last Tenor 1Y
Interpolate Outright Variance true
Interpolate on Trading Time false
DateCut of Expiries NYC
Quotes are Delta with Premium false
Up Extrap 1.0 Delta 3.0
Down Extrap 1.0 Delta 0.0
Granularity Continuous
Weighting false
FXDate 06/05/2009
Roll Calendar Vol false
Comment
Same data as "Volatility Testing 2", using 1Y test data from spreadsheet E1.
Load... New Delete... Save Save As Close


FXVolatilitySurface Heston Testing E1 1Y USD JPY User(calypso_user)
Surface Utilities Help
Name Heston Testing E1 1Y CLOSE Date 06/05/2009 7:48:46 AM
Definition Underlyings Quotes Points Graph
Quote Name Type CLOSE Weight
FXOption.USD/JPY.1Y.ATM Yield 14.48000000 1.00000
FXOption.USD/JPY.1Y.Butterfly.25-delta Yield 0.57000001 1.00000
FXOption.USD/JPY.1Y.Risk Reversal.25-delta Yield -5.00000000 1.00000
FXOption.USD/JPY.1Y.Butterfly.10-delta Yield 2.30000000 1.00000
FXOption.USD/JPY.1Y.Risk Reversal.10-delta Yield -9.79000000 1.00000



9.4.2 Differential Evolution and its Parameters

Differential Evolution is a form of genetic algorithm which takes a population of trial parameter sets and incrementally improves the population over successive generations until a target condition is satisfied. In the Heston calibration, the target function is the sum of the squared errors -- that is, for each trial set of Heston parameters the vanilla options are priced, each price is compared to the market quoted premium, the difference taken and squared, and those squares summed. The search continues until that target function reaches a minimum value, or until no improvement is made, or until a maximum number of generations has been reached.

The vanilla options are priced in the Heston model using the well-known semi-analytic expression, taking care with the complex logarithm and speeding the procedure by eliminating redundant calculations on options of the same expiry.

The generator FXOptionDeltaHeston has several generation parameters to control the calibration, in addition to the usual parameters found in FXOptionDelta. The new generation parameters are as follows:

Fixed volatility : A volatility (Heston initial volatility) that may be input instead of calibrated. This will speed the calibration as it reduces the number of free variables. Since this value usually changes slowly from day to day it is useful to input the value from a previous day's calibration.

Fixed rev speed : Similar to "Fixed volatility," the reversion speed is known in practice to change slowly over time, so it need not be calibrated every day. A previously calibrated value can be placed here.

DE - Population size : The number of parameter sets tested for one generation. Each parameter set produces one value of the target function (sum of squared errors).

DE - Max num generations : The maximum number of generations allowed before the algorithm is forced to terminate. This ensures termination of the program in a desired time interval. The total number of target function valuations is the Population size times the Max number of generations.

DE - F : The scale parameter of Differential Evolution, between 0 and 1; used in rescaling differences between trial parameter sets.

DE - CR : The crossover acceptance rate of Differential Evolution, between 0 and 1, determining how often the trial set is randomly accepted over the original set.

The Differential Evolution parameters are set with reasonable default values, so it is not necessary to understand all parameters in order to effectively perform the calibration. It is recommended that one try using small values for Population size and Max num generations when experimenting with the calibration in order to see fast results.

9.4.3 Log Category

To see the evolution algorithm at work, use the Log category **"DiffEv"** while performing a calibration.

9.4.4 Timing

The time taken to perform a Heston calibration will vary greatly depending on the number of vanilla option strikes and expiries, and the chosen population size and number of generations. Very large numbers of options with highly accurate calibrations can take up to 15 minutes to produce. Calypso's times for this calibration match the best times that have been published in the literature.

As an example, calibrating the model to 32 options at 20 expiries (640 total option prices) resulted in the algorithm finding an acceptable solution in 2.8 minutes. The target function value of the optimal parameter set was on the order of 0.0001. This employed 400 target evaluations, or a total of 400 * 32 * 20 = Heston vanilla option price evaluations. Note that with a Population size of 20 and a Max number of generations of 50, a total of 1000 target evaluations (640,000 option pricings) could have been performed before the algorithm was forced to halt; the fact that the algorithm halted sooner meant that it could not find an improvement after several attempts.

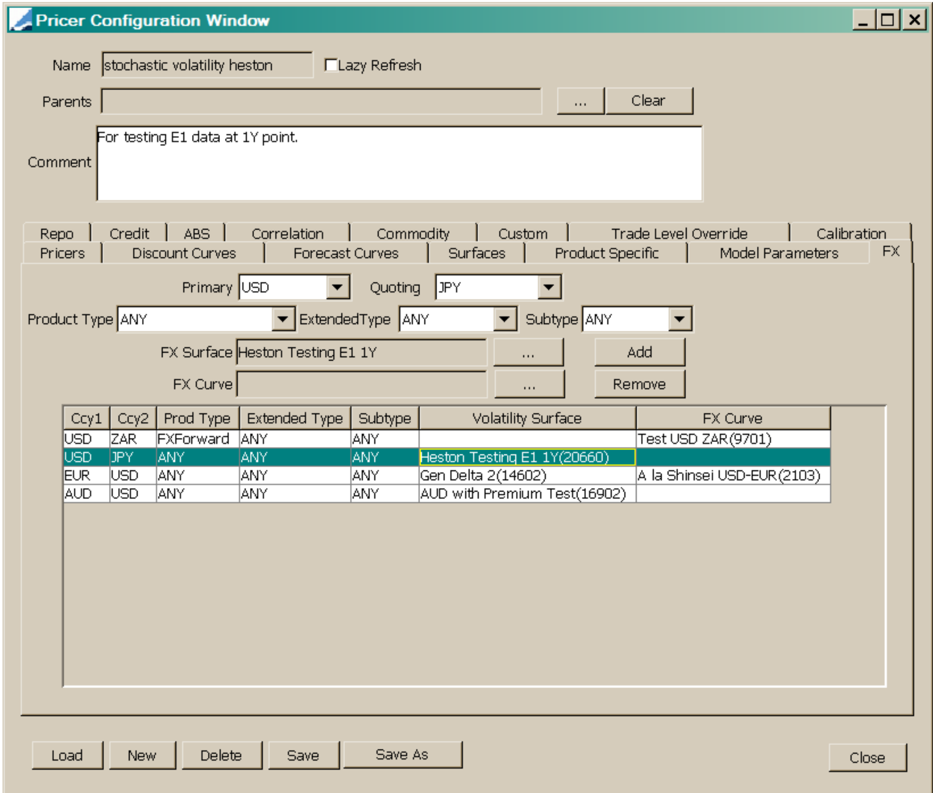
9.4.5 Lack of Uniqueness

The calibration of the Heston model is not unique. Quite a variety of parameter sets can give essentially the same results for the target function. Unless one places additional criteria on the result, there is no reason to prefer one solution set over another. Therefore one should not become overly concerned with each Heston parameter value itself, which could be very dependent on the calibration choices, as it is the joint action of the parameters in producing the final results which is important.

9.5 Configuring a Pricer to Use a Calibration

Set up a Pricer Configuration to use an FX volatility surface that has been generated using FXOptionDeltaHeston. then one of the pricers that use the Heston model will pick up the Heston parameters from this surface, so that the Pricing Parameters do not need to be entered manually in the trade window. The same surface also can be used as a standard FX Volatility Surface, so that any pricer that does not use the Heston model can still use this surface to look up volatilities as usual.

Note: Proper use of this configuration requires an enhancement that was performed after Release 11.0.



Pricer Configuration Window
Name
stochastic volatility heston
Lazy Refresh
Parents
...
Clear
Comment
For testing E1 data at 1Y point.
Repo
Credit
ABS
Correlation
Commodity
Custom
Trade Level Override
Calibration
Pricers
Discount Curves
Forecast Curves
Surfaces
Product Specific
Model Parameters
FX
Primary
USD
Quoting
JPY
Product Type
ANY
ExtendedType
ANY
Subtype
ANY
FX Surface
Heston Testing E1 1Y
...
Add
FX Curve
...
Remove
Ccy1
Ccy2
Prod Type
Extended Type
Subtype
Volatility Surface
FX Curve
USD ZAR FXForward ANY ANY Test USD ZAR(9701)
USD JPY ANY ANY ANY Heston Testing E1 1Y(20660)
EUR USD ANY ANY ANY Gen Delta 2(14602) A la Shinsei USD-EUR(2103)
AUD USD ANY ANY ANY AUD with Premium Test(16902)
Load
New
Delete
Save
Save As
Close

Section 10. Money Market Instruments

PricerSimpleMM is an empty class. PricerIntraDayMM and PricerCashCommodity extend PricerSimpleMM and only implement some back office specific functions. PricerCallNotice extends PricerCash and implements its own method to calculate accrual, but there is only some implementation difference, not analytically. Hence, PricerSimpleMM is the only pricer class needed to be explored.

10.1.1 PricerSimpleMM

SimpleMM Pricer evaluates either fixed rate or floating rate loan/deposit. Keep in mind it's from bank's prospective. So the first cash flow usually is negative if it's a loan. When evaluating floating SimpleMM, an index rate and a spread should be specified. If the interest payment is due on the start date, the Discount check box needs to be selected.

In PricerSimpleMM's process method, following analytic measures are calculated:

- NPV/Price/Marginal Call
- B/E rate
- Accrual/Accrual_BC/Accrual_First/Indemnity_Accrual
- PV01
- Fees_NPV

Pricing Measures

Following is the description of some measures.

PRICE — "Price" denotes the Market Price of the security.

NPV — The Net Present Value (NPV) of a series of cash flows is the sum of the present values of each of the cash flows, some or all of which may be negative.

$$NPV = \sum_{i=0}^{N-1} C_i \times df_i$$

- N is total number of cash flow
- C_i is the *i*th cash flow
- df_i is the *i*th discount factor corresponding to the *i*th cash flow

NPV uses Trade, SimpleMM, PricerSimpleMMInput, PricingEnv, PricerMeasure classes. An integer curveSide, a Boolean to indicate if include fee and the valuation date.

The curveSide integer in process function has been settled to QuoteSet.MID, which means the discount curve and forecast curve will use the middle value of the ask and bid prices of quotes.

Process delegates to NPV calculation task to function computeNPV, which in turn delegates the task to function pvFixed and function pvFloating to calculate fixed rate SimepleMM and floating rate Simple MM.

In pvFixed, it first checks if the inputted Trade object has the valid CashFlowSet, if not, it let the object to generate and calculate cashflows. (Details of cashflow generation and calculation see swap and bond pricer documents) Then in a loop, pvFixed accumulatesd the present value of each cash flow. The discount factor is retrieved from the discount curve in the PricingEnv object and the cash flow amount is retrieved using a helper function getCashFlowAmount.

In pvFloating, the only difference from pvFixed is that in the loop the cash flow amount is retrieved using a helper function forecastFlow.

ACCRUAL — Accrued Interest is the proportion of interest or coupon earned on an investment from the previous coupon payment date until the value date.

ACCRUAL_FIRST — "Accrual First" denotes the linear accrual including the day of calculation.

CASH — "Cash" denotes the sum of all the cash flows occurring on the Valuation Date.

NOTIONAL — In a Bond Futures contract, the bond bought or sold is a standardized, non-existent, notional bond, as opposed to the actual bonds that are deliverable at maturity. Contracts for differences also require a notional principal amount on which settlement amount can be calculated.

ACCRUAL_BO — Apart from Bonds, Accrual_BO is the same as Accrual. For Bonds, Accrual_BO is the Accrual computed on Valuation Date for same Date (as opposed to the Accraul of the Bond that computes the value adjusted for the settle days).

B/E_Rate — The B/E Rate, or the Break Even rate is the average Interest rate at which a zero profit is recorded. In our system, this would be represented by an NPV = 0.

Break even rate is an interest/coupon rate that makes the NPV of the simple money market zero. In the case of fixed rate simple money market, it’s the fixed rate which makes the NPV zero.

$$\sum_{i=0}^{N-1}(P \times BVR \times df_i) + P \times df_{N-1} - P = 0$$

- N is the total number of cash flow
- P is the notional principle
- BVR is the bireak even rate
- df_i is the ith discount factor

In the case of floating rate simple money market, break even rate is the spread.

$$\sum_{i=0}^{N-1}\left[P \times (IR_i + BVR) \times df_i\right] + P \times df_{N-1} - P = 0$$

- IR is the index rate

Break Even Rate uses function solveForBreakEvenRate and class SolverSimpleMM.

SolverSimpleMM’s solve function implement an iterative algorithm to solve the equation of BER described in algorithm section.

PV01 — "Present Value of an 01 (PV01)" or "Dollar Value of an 01 (DV01)" or "the value of a basis point" is the change in price due to a 1 basis point change in yield. This is usually expressed as appositive number.

There are also several back office measures such as Funding_MTM, etc.

Pricer Parameters

- MMKT_From_Quote
  - Values — True of False.
  - Usage — When set to true, the product would be priced from quote.
- Instance_Type
  - Values — Close, Open, Last.
  - Usage: Determines the type of quote to use in quote set, for example, when Close is chosen, close quote will be used.
- Repo_rate
  - Values — Double.
  - Usage — If the valuation date is before the settle date , Pricer 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 pricer 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 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_Including_Cash
  - Values — True or False.
  - Usage — If set to true, and if the there is cash flow on valuation date, the pricer will add/subtract the cash.
- NPV_Including_Cost
  - Values — True or False.
  - Usage — If set to true, the NPV would reflect settlement cost.

Section 11. Bond and Money Market Futures

Product	Calypso Product	Description	Pricer
Bond Future	FutureBond		PricerFutureBond
Money Market Future	FutureMM		PricerFutureMM
Bond Future Option	FutureOptionBond		PricerFutureOptionBond
Money Market Future Option	FutureOptionMM		PricerFutureOptionMM

11.1 PricerFutureBond

NPV — General Formula

Future Price = Forward Bond Price / Cheapest To Deliver Factor. The Forward Bond Price is the price of the selected cheapest to deliver bond at the future delivery date (including accrual adjustment which accounts for those coupons that fall between the valuation date and the future delivery date). The cheapest to deliver factor accounts for the difference between the actual bond used at delivery, and the synthetic 8%, 15 year non-callable benchmark bond.

11.2 PricerFutureMM

NPV — General Formula

Future Price = 100 – forward yield. The forward yield is derived from the zero curve by calculating the (annually compounded) forward rate between the maturity date of the futures contract and the maturity date of the rate underlying the futures contract. Currently we have not implemented the Eurodollar convexity adjustment which is given by:

$$\Delta Futures\,\Pr ice = -0.5\sigma^2 t_1 t_2$$

With t_1 being the maturity date of the futures contract

And t_2 being the maturity date of the rate underlying the futures contract

11.3 PricerFutureOptionBond and PricerFutureOptionMM

NPV — General Formula European Style

This pricer implements the Black Scholes formula as follows:

- Formula for call

  $c = e^{-rT}(FN(d_1) - X\,N(d_2))$

  With

  $d_1 = \dfrac{\ln\left(\dfrac{F}{X}\right) + \dfrac{\sigma^2}{2}T}{\sigma\sqrt{T}}$,

  $d_2 = d_1 - \sigma\sqrt{T}$,

  T : Time Period from valuation date to exercise date,

F : Futures price

X : Strike

- Formula for put:

$$p = e^{-rT}(XN(-d_2) - FN(-d_1))$$

- Greeks: Greeks are defined as follows with p being the npv of the call or put, respectively:

$$Delta = \frac{\partial p}{\partial F}$$

$$Gamma = \frac{\partial^2 p}{\partial F^2}$$

$$Vega = \frac{\partial p}{\partial \sigma}$$

$$Theta = \frac{\partial p}{\partial \tau}$$

$$Rho = \frac{\partial p}{\partial r}$$

NPV — General Formula American Style

This pricer implements currently a binomial (recombining tree). Details of this standard implementation can be found in *Options, Futures and Other Derivative Securities by John Hull 4th Edition (2000)*, pp.213. In the near future, an implied trinomial tree for faster convergence will replace this binomial tree. The numerical results will be the same but the computation speed will still further improve. The Greeks (Delta, Gamma, Vega, Theta, Rho) are computed numerically and scaled the same way so that they exactly agree with the Greeks of the Black-Scholes formula. The results of a European option evaluated on the tree and with Black-Scholes are consistent if the number of steps is 1000 or more.

11.4 Future Option Volatility Surfaces

11.4.1 Future Option Generator

Future Option Generator — Underlying instruments (money market future option, bond future option) are quoted as prices. Generates an implied volatility surface.