Adenza

Calypso CALIB FXO Pricers

Version 17.0

February 2022 – First Edition

This document exposes the scope and underlying assumptions of the FXO pricers available in Calypso's Calibmodule.

Revision Date	Comments
February 2022	First edition for version 17.0.

Contents

Section 1. FXO Pricers

The Calib FXO Pricers give access to the latest analytics for valuation of FX derivatives developed by the Financial Engineering team at Calypso.

The FXO pricers have been designed to serve as a replacement of, and an improvement over, Calypso classic FXOption pricers.

As a hint to help distinguish newer Calib pricers, from classic pricers, Calib pricers use abbreviated FXO particle within their name, while classic pricers use full FXOption particle.

The inclusion of these pricers in Calib creates the benefit of efficient installation and configuration of the latest FXO analytics without the need in many cases of upgrading the core Calypso platform.

Calib is the Calypso Analytics Library, a Calypso module that contains analytics. Further documentation in CalLib can be found in the Calypso Analytics Library guide.

The Calib FXO Pricers library is compatible with Calypso version 12.0.0.0 onwards.

Note that not all pricing functionality will be accessible in all supported Calypso versions. Only products, or flavors, available in a particular version, can be valuated with FXO pricers.

Section 2. Product coverage

Next table summarizes option coverage of the different FXO pricers. Provided that there are no errors in the tables, the meaning is as follows:

Any No, indicates that the respective combination of Pricer and Option type is not supported and should not be used in production, not even in the event that numbers are produced for such a combination (that would indicate a bug in the constrains checker).

Any Yes, indicates that the indicated combination of Pricer and Option type is fully supported under the assumptions of such a pricer. That is by no means a guarantee of the good performance of the model in different aspects like matching to market, hedging hints or behavior when employed for other purposes apart from valuation and calculation of hedging measures (for instance, the usage of these models for VAR purposes could produce strange results).

However, misbehavior of the model, unstability of the results in a way that can prevent normal usage, or speed clearly below what is expected from the particular model, will be considered as a deviation from the documentation from the responsible engineers, and addressed promptly and accordingly with the prevailing Calypso policy about it.

2.1 FX Option - Vanilla

Exercise Type	Settlement Type	PricerFXO									
		Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
European	**[Late] Physical**	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
European	**[Late] Cash**	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
European	**[Late] 3rd Ccy Flexo**	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
European	**[Late] Self-Quanto**	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
European	**[Late] 3rd Ccy Quanto**	Yes	No	No	No	Yes	No	Yes	No	No	No
American	**[Late] Physical**	Yes	No	No	Yes (*)	No	No	No	No	No	No
American	**[Late] Cash**	Yes	No	No	Yes (*)	No	No	No	No	No	No
American	**[Late] 3rd Ccy Flexo**	Yes	No	No	Yes (*)	No	No	No	No	No	No
American	**[Late] Self-Quanto**	Yes	No	No	No	No	No	No	No	No	No
American	**[Late] 3rd Ccy Quanto**	Yes	No	No	No	No	No	No	No	No	No

(*) The valuation of American vanillas under the Market pricer, is not strictly possible under its modelling assumptions. However, good modeling of this type of product is highly sophisticated, and so a *just better than Theoretical* implementation has been made available under this pricer. See notes respective to PricerFXOMarket for more information.

2.2 FX Option – FwdStart

Strike Format	Settlement Type	PricerFXO									
		Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
% atms, % otm	[Late] Physical	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
% atms, % otm	[Late] Cash	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
% atms, % otm	[Late] 3rd Ccy Flexo	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
% atms, % otm	[Late] Self-Quanto	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
% atms, % otm	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	No	No	No	No
% atmf	[Late] Physical	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
% atmf	[Late] Cash	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
% atmf	[Late] 3rd Ccy Flexo	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
% atmf	[Late] Self-Quanto	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
% atmf	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	No	No	No	No
pips	[Late] Physical	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
pips	[Late] Cash	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
pips	[Late] 3rd Ccy Flexo	Yes	No	No	Yes (*)	Yes	Yes	No	No	No	No
pips	[Late] Self-Quanto	Yes	No	No	Yes (*/ **)	Yes	Yes	No	No	No	No
pips	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	No	No	No	No

(*) The valuation of Fwd Start vanillas under the Market pricer, is not strictly possible under its modelling assumptions. However, good modeling of this type of product is highly sophisticated, and so a *just better than TV* implementation has been made available under this pricer. See notes respective to PricerFXOMarket for more information.

(**) Valuation of pips-style Self-Quanto by FXOMarket pricer is slow. Contact Calypso FE team if that becomes aproblem.

2.3 FX Option – Digital

Trigger Duration	Trigger Type	Payout Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
EXPIRY	ABOVE	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	ABOVE	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes	No	No	No
EXPIRY	BELOW	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	BELOW	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes	No	No	No
EXPIRY	OUT	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	OUT	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes	No	No	No
EXPIRY	IN	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	IN	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes	No	No	No
FULL	NT UP, NT DN	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	NT UP, NT DN	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	OT UP, OT DN	Expiry	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	OT UP, OT DN	Expiry	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	OT UP, OT DN	Instant	Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	OT UP, OT DN	Instant	3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	DNT	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DNT	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	DOT	Expiry	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DOT	Expiry	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	DOT	Instant	Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DOT	Instant	3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	OTNT (UI)	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	OTNT (UI)	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	OTNT (DI)	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes

Trigger Duration	Trigger Type	Payout Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
FULL	OTNT (DI)	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
PARTIAL	NT UP, NT DN	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	NT UP, NT DN	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
PARTIAL	OT UP, OT DN	Expiry	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	OT UP, OT DN	Expiry	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
PARTIAL	OT UP, OT DN	Instant	Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	OT UP, OT DN	Instant	3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
PARTIAL	DNT	#N/A	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DNT	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	DOT	Expiry	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DOT	Expiry	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	DOT	Instant	Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DOT	Instant	3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No

Trigger Duration	Trigger Type	Payout Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
PARTIAL	OTNT(UI)	#N/A	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	OTNT(UI)	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	OTNT(DI)	#N/A	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	OTNT(DI)	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No

(*) See explanation under FXOTheoretical section of limitations on FULL and PARTIAL analytical formulas forbarriers.

(**) By now, PARTIAL double barriers are only managed when the duration of both barriers is the same.

2.4 FX Option – Barrier

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
EXPIRY	UO, DO	#N/A	[Late] Physical	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UO, DO	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UO, DO	#N/A	[Late] 3rd Ccy Flexo	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UO, DO	#N/A	[Late] Self-Quanto	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UO, DO	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes (***)	No	No	No
EXPIRY	UI, DI	#N/A	[Late] Physical	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
EXPIRY	UI, DI	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UI, DI	#N/A	[Late] 3rd Ccy Flexo	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UI, DI	#N/A	[Late] Self-Quanto	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UI, DI	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes (***)	No	No	No
EXPIRY	DKO	#N/A	[Late] Physical	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKO	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKO	#N/A	[Late] 3rd Ccy Flexo	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKO	#N/A	[Late] Self-Quanto	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKO	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes (***)	No	No	No
EXPIRY	DKI	#N/A	[Late] Physical	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKI	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKI	#N/A	[Late] 3rd Ccy Flexo	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKI	#N/A	[Late] Self-Quanto	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKI	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes (***)	No	No	No
EXPIRY	UIDO	#N/A	[Late] Physical	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UIDO	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UIDO	#N/A	[Late] 3rd Ccy Flexo	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UIDO	#N/A	[Late] Self-Quanto	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UIDO	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes (***)	No	No	No
EXPIRY	DOUI	#N/A	[Late] Physical	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DOUI	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DOUI	#N/A	[Late] 3rd Ccy Flexo	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DOUI	#N/A	[Late] Self-Quanto	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DOUI	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes (***)	No	No	No

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
EXPIRY	KIKO (UI)	#N/A	[Late] Physical	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (UI)	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (UI)	#N/A	[Late] 3rd Ccy Flexo	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (UI)	#N/A	[Late] Self-Quanto	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (UI)	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes	No	No	No
EXPIRY	KIKO (DI)	#N/A	[Late] Physical	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (DI)	#N/A	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (DI)	#N/A	[Late] 3rd Ccy Flexo	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (DI)	#N/A	[Late] Self-Quanto	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (DI)	#N/A	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes	No	No	No
FULL	UO, DO	Expiry	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UO, DO	Expiry	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UO, DO	Expiry	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UO, DO	Expiry	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UO, DO	Expiry	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes (***)	No	No	No
FULL	UO, DO	Instant	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UO, DO	Instant	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UO, DO	Instant	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UO, DO	Instant	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UO, DO	Instant	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes (***)	No	No	No

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
FULL	UI,DI	#N/A	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UI,DI	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UI,DI	#N/A	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UI,DI	#N/A	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UI,DI	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes (***)	No	No	No
FULL	DKO	Expiry	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKO	Expiry	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKO	Expiry	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKO	Expiry	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKO	Expiry	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes (***)	No	No	No
FULL	DKO	Instant	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKO	Instant	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKO	Instant	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKO	Instant	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKO	Instant	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes (***)	No	No	No
FULL	DKI	#N/A	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKI	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKI	#N/A	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKI	#N/A	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
FULL	DKI	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes (***)	No	No	No
FULL	UIDO	#N/A	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UIDO	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UIDO	#N/A	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UIDO	#N/A	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UIDO	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	UODI	#N/A	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UODI	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UODI	#N/A	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UODI	#N/A	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UODI	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	KIKO (UI)	#N/A	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	KIKO (UI)	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	KIKO (UI)	#N/A	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	KIKO (UI)	#N/A	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	KIKO (UI)	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	KIKO (DI)	#N/A	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	KIKO (DI)	#N/A	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	KIKO (DI)	#N/A	[Late] 3rd Ccy Flexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	KIKO	#N/A	[Late] Self-Quanto	Yes	Yes	No	No	No	No	Yes	Yes	Yes	Yes

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
	(DI)			(*)	(*)								
FULL	KIKO (DI)	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
PARTIAL	UO, DO	Expiry	[Late] Physical	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UO, DO	Expiry	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UO, DO	Expiry	[Late] 3rd CcyFlexo	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UO, DO	Expiry	[Late] Self-Quanto	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UO, DO	Expiry	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes (***)	No	No	No
PARTIAL	UO, DO	Instant	[Late] Physical	Yes (*)	Yes(*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UO, DO	Instant	[Late] Cash	Yes (*)	Yes(*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UO, DO	Instant	[Late] 3rd CcyFlexo	Yes (*)	Yes(*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UO, DO	Instant	[Late] Self-Quanto	Yes (*)	Yes(*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UO, DO	Instant	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes (***)	No	No	No
PARTIAL	UI,DI	#N/A	[Late] Physical	Yes (*)	Yes(*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UI,DI	#N/A	[Late] Cash	Yes (*)	Yes(*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UI,DI	#N/A	[Late] 3rd CcyFlexo	Yes (*)	Yes(*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UI,DI	#N/A	[Late] Self-Quanto	Yes (*)	Yes(*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UI,DI	#N/A	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes (***)	No	No	No
PARTIAL	DKO	Expiry	[Late] Physical	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
PARTIAL	DKO	Expiry	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKO	Expiry	[Late] 3rd CcyFlexo	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKO	Expiry	[Late] Self-Quanto	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKO	Expiry	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes (***)	No	No	No
PARTIAL	DKO	Instant	[Late] Physical	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKO	Instant	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKO	Instant	[Late] 3rd CcyFlexo	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKO	Instant	[Late] Self-Quanto	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKO	Instant	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes (***)	No	No	No
PARTIAL	DKI	#N/A	[Late] Physical	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKI	#N/A	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKI	#N/A	[Late] 3rd CcyFlexo	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKI	#N/A	[Late] Self-Quanto	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKI	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes (***)	No	No	No

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
PARTIAL	UIDO	#N/A	[Late] Physical	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UIDO	#N/A	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UIDO	#N/A	[Late] 3rd CcyFlexo	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UIDO	#N/A	[Late] Self-Quanto	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UIDO	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	UODI	#N/A	[Late] Physical	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UODI	#N/A	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UODI	#N/A	[Late] 3rd CcyFlexo	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UODI	#N/A	[Late] Self-Quanto	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UODI	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	KIKO (UI)	#N/A	[Late] Physical	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	KIKO (UI)	#N/A	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	KIKO (UI)	#N/A	[Late] 3rd CcyFlexo	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	KIKO (UI)	#N/A	[Late] Self-Quanto	Yes (*/	Yes (*/	No	No	No	No	Yes	Yes	Yes	Yes

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
				**)	**)								
PARTIAL	KIKO (UI)	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	KIKO (DI)	#N/A	[Late] Physical	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	KIKO (DI)	#N/A	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	KIKO (DI)	#N/A	[Late] 3rd CcyFlexo	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	KIKO (DI)	#N/A	[Late] Self-Quanto	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	KIKO (DI)	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
MULTI_ PERIOD	UO,DO	Expiry	[Late] Physical	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UO,DO	Expiry	[Late] Cash	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UO,DO	Expiry	[Late] 3rd CcyFlexo	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UO,DO	Expiry	[Late] Self-Quanto	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UO,DO	Expiry	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes (***)	No	No	No
MULTI_ PERIOD	UO,DO	Instant	[Late] Physical	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UO,DO	Instant	[Late] Cash	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
MULTI_ PERIOD	UO,DO	Instant	[Late] 3rd CcyFlexo	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UO,DO	Instant	[Late] Self-Quanto	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UO,DO	Instant	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes (***)	No	No	No
MULTI_ PERIOD	UI,DI	#N/A	[Late] Physical	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UI,DI	#N/A	[Late] Cash	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UI,DI	#N/A	[Late] 3rd CcyFlexo	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UI,DI	#N/A	[Late] Self-Quanto	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	UI,DI	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes (***)	No	No	No
MULTI_ PERIOD	DKO	Expiry	[Late] Physical	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKO	Expiry	[Late] Cash	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKO	Expiry	[Late] 3rd CcyFlexo	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKO	Expiry	[Late] Self-Quanto	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKO	Expiry	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes (***)	No	No	No
MULTI_ PERIOD	DKO	Instant	[Late] Physical	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes

Barrier Duration	Barrier Type	Rebate Type	Settle Type	PricerFXO									
				Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
MULTI_ PERIOD	DKO	Instant	[Late] Cash	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKO	Instant	[Late] 3rd CcyFlexo	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKO	Instant	[Late] Self-Quanto	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKO	Instant	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes (***)	No	No	No
MULTI_ PERIOD	DKI	#N/A	[Late] Physical	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKI	#N/A	[Late] Cash	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKI	#N/A	[Late] 3rd CcyFlexo	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKI	#N/A	[Late] Self-Quanto	Yes (*/ **)	No	No	No	No	No	Yes	Yes	Yes	Yes
MULTI_ PERIOD	DKI	#N/A	[Late] 3rd Ccy Quanto	Yes (*/ **)	No	No	No	No	No	Yes (***)	No	No	No

(*) See explanation under FXOTheoretical section of limitations on FULL and PARTIAL analytical formulas for barriers.

(**) PARTIAL and MULTI_PERIOD double barriers are valuated with analytical formulas under FXOTheoretical only when the count of distinct relevant dates is 3 or less. The relevant dates for PARTIAL double barriers are: Up Barrier Start Date, Up Barrier End Date, Down Barrier Start Date, End Barrier Start Date, Expiry Date. The relevant dates for MULTI_PERIOD barriers are: the Start Date and End Date of every period, and the Expiry Date. If more than 3 distinct dates are detected, FXOTheoreticalFiniteDifference will be used for valuation.

(***) Quanto barriers not managed if the rebate is not zero.

2.5 FX Option – Digital with Barrier

Barrier Duration	Barrier Type	Settle Type	PricerFXO									
			Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
EXPIRY	UO, DO	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UO, DO	[Late] 3rd Ccy Quanto	Yes	Yes	No	No	Yes	No	Yes	No	No	No
EXPIRY	UI, DI	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UI, DI	[Late] 3rd Ccy Quanto	Yes	Yes	No	No	Yes	No	Yes	No	No	No
EXPIRY	DKO	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKO	[Late] 3rd Ccy Quanto	Yes	Yes	No	No	Yes	No	Yes	No	No	No
EXPIRY	DKI	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DKI	[Late] 3rd Ccy Quanto	Yes	Yes	No	No	Yes	No	Yes	No	No	No
EXPIRY	UIDO	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	UIDO	[Late] 3rd Ccy Quanto	Yes	Yes	No	No	Yes	No	Yes	No	No	No
EXPIRY	DOUI	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	DOUI	[Late] 3rd Ccy Quanto	Yes	Yes	No	No	Yes	No	Yes	No	No	No
EXPIRY	KIKO (UI)	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (UI)	[Late] 3rd Ccy Quanto	Yes	Yes	No	No	Yes	No	Yes	No	No	No
EXPIRY	KIKO (DI)	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EXPIRY	KIKO (DI)	[Late] 3rd Ccy Quanto	Yes	Yes	No	No	Yes	No	Yes	No	No	No
FULL	UO, DO	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UO, DO	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	UI, DI	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UI, DI	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	DKO	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	DKO	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	DKI	[Late] Cash	Yes (*)	Yes	No	No	No	No	Yes	Yes	Yes	Yes

Barrier Duration	Barrier Type	Settle Type	PricerFXO									
			Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
				(*)								
FULL	DKI	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	UIDO	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UIDO	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	UODI	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	UODI	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	KIKO (UI)	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	KIKO (UI)	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
FULL	KIKO (DI)	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
FULL	KIKO (DI)	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
PARTIAL	UO, DO	[Late] Cash	Yes (*)	Yes (*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UO, DO	[Late] 3rd CcyQuanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
PARTIAL	UI, DI	[Late] Cash	Yes (*)	Yes(*)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UI, DI	[Late] 3rd CcyQuanto	Yes (*)	No	No	No	No	No	Yes	No	No	No
PARTIAL	DKO	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKO	[Late] 3rd CcyQuanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	DKI	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	DKI	[Late] 3rd CcyQuanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	UIDO	[Late] Cash	Yes (*/	Yes (*/	No	No	No	No	Yes	Yes	Yes	Yes

Barrier Duration	Barrier Type	Settle Type	PricerFXO									
			Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
			**)	**)								
PARTIAL	UIDO	[Late] 3rd CcyQuanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	UODI	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	UODI	[Late] 3rd CcyQuanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	KIKO (UI)	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	KIKO (UI)	[Late] 3rd CcyQuanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No
PARTIAL	KIKO (DI)	[Late] Cash	Yes (*/ **)	Yes (*/ **)	No	No	No	No	Yes	Yes	Yes	Yes
PARTIAL	KIKO (DI)	[Late] 3rd CcyQuanto	Yes (*/ **)	No	No	No	No	No	Yes	No	No	No

(*) See explanation under FXOTheoretical section of limitations on FULL and PARTIAL analytical formulas for barriers.

(**) PARTIAL double barriers are valuated with analytical formulas under FXOTheoretical only when the count of distinct relevant dates is 3 or less. The relevant dates are: Up Barrier Start Date, Up Barrier End Date, Down Barrier Start Date, End Barrier Start Date, Expiry Date. If more than 3 distinct dates are detected, FXOTheoreticalFiniteDifference will be used for valuation.

2.6 FX Option – Accrual

Observation type	Level type	Settle Type	PricerFXO Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
Cash Accrual	Known	[Late] Cash	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Cash Accrual	Known	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes	No	No	No
Cash Accrual	Resettable %	[Late] Cash	Yes	No	No	Yes	Yes	Yes	No	No	No	No
Cash Accrual	Resettable %	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	No	No	No	No
Cash Accrual	Resettable pips	[Late] Cash	Yes	No	No	Yes	Yes	Yes	No	No	No	No
Cash Accrual	Resettable pips	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	No	No	No	No
FX Accrual	Known	[Late] Physical	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
FX Accrual	Resettable %	[Late] Physical	Yes	No	No	Yes	Yes	Yes	No	No	No	No
FX Accrual	Resettable pips	[Late] Physical	Yes	No	No	Yes	Yes	Yes	No	No	No	No
Vanilla Fade In, Vanilla Fade Out	Known	[Late] Physical	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Vanilla Fade In, Vanilla Fade Out	Known	[Late] Cash	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Vanilla Fade In, Vanilla Fade Out	Known	[Late] 3rd Ccy Flexo	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Vanilla Fade In, Vanilla Fade Out	Known	[Late] Self-Quanto	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Vanilla Fade In, Vanilla Fade Out	Known	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	Yes	No	No	No

2.7 FX Option – Asian

Observation type	Settlement type	PricerFXO									
		Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
Average Rate	[Late] Cash in Quoting	Yes (*)	No	No	Yes (*)	Yes	Yes	No	No	No	No
Average Rate	[Late] Cash in Primary	Yes (*)	No	No	No	Yes	Yes	No	No	No	No
Average Rate	[Late] 3rd Ccy Flexo	Yes (*)	No	No	Yes (*)	Yes	Yes	No	No	No	No
Average Rate	[Late] Self-Quanto	Yes (*)	No	No	No	Yes	Yes	No	No	No	No
Average Rate	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	Yes	No	No	No	No	No
Geom Avg Rate	[Late] Cash in Quoting	Yes	No	No	Yes	Yes	Yes	No	No	No	No
Geom Avg Rate	[Late] Cash in Primary	Yes	No	No	No	Yes	Yes	No	No	No	No
Geom Avg Rate	[Late] 3rd Ccy Flexo	Yes	No	No	Yes	Yes	Yes	No	No	No	No
Geom Avg Rate	[Late] Self-Quanto	Yes	No	No	No	Yes	Yes	No	No	No	No
Geom Avg Rate	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	No	No	No	No
Average Strike	[Late] Physical	Yes (*)	No	No	Yes (*/ **)	Yes	Yes	No	No	No	No
Average Strike	[Late] Cash	Yes (*)	No	No	Yes (*/ **)	Yes	Yes	No	No	No	No
Average Strike	[Late] 3rd Ccy Flexo	Yes (*)	No	No	Yes (*/ **)	Yes	Yes	No	No	No	No
Average Strike	[Late] Self-Quanto	Yes (*)	No	No	Yes (*/ **)	Yes	Yes	No	No	No	No
Average Strike	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	Yes	No	No	No	No	No
Geom Avg Strike	[Late] Physical	Yes	No	No	Yes (**)	Yes	Yes	No	No	No	No
Geom Avg Strike	[Late] Cash	Yes	No	No	Yes (**)	Yes	Yes	No	No	No	No
Geom Avg Strike	[Late] 3rd Ccy Flexo	Yes	No	No	Yes (**)	Yes	Yes	No	No	No	No
Geom Avg Strike	[Late] Self-Quanto	Yes	No	No	Yes (**)	Yes	Yes	No	No	No	No
Geom Avg Strike	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	No	No	No	No
DARO	[Late] Cash in Quoting	Yes (*)	No	No	Yes (*/ **/	Yes	Yes	No	No	No	No

Observation type	Settlement type	PricerFXO									
		Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
					***)						
DARO	[Late] Cash in Primary	Yes (*)	No	No	No	Yes	Yes	No	No	No	No
DARO	[Late] 3rd Ccy Flexo	Yes (*)	No	No	Yes (*/ **/ ***)	Yes	Yes	No	No	No	No
DARO	[Late] Self-Quanto	Yes (*)	No	No	No	Yes	Yes	No	No	No	No
DARO	[Late] 3rd Ccy Quanto	Yes (*)	No	No	No	Yes	No	No	No	No	No
Geom DARO	[Late] Cash in Quoting	Yes	No	No	Yes (**/ ***)	Yes	Yes	No	No	No	No
Geom DARO	[Late] Cash in Primary	Yes	No	No	No	Yes	Yes	No	No	No	No
Geom DARO	[Late] 3rd Ccy Flexo	Yes	No	No	Yes (**/ ***)	Yes	Yes	No	No	No	No
Geom DARO	[Late] Self-Quanto	Yes	No	No	No	Yes	Yes	No	No	No	No
Geom DARO	[Late] 3rd Ccy Quanto	Yes	No	No	No	Yes	No	No	No	No	No

(*) See explanation under FXOTheoretical section of limitations on analytical formulas for arithmetic asian. (**) Valuation of average strike or double average Asian options under Market model is relatively slow.

(***) Market model can be applied to double average Asian options only when the first fixing of the average rate, is occurring after (or exactly at the same time) the last fixing of the average strike.

2.8 FX Option – Lookback

Observation type	Settlement type	PricerFXO									
		Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
Lookback Rate	[Late] Cash in Quoting	Yes (*)	No	No	No	Yes	Yes	No	No	No	No
Lookback Rate	[Late] Cash in Primary	Yes (*)	No	No	No	Yes	Yes	No	No	No	No
Lookback Strike	[Late] Physical	Yes (*)	No	No	No	Yes	Yes	No	No	No	No
Lookback Strike	[Late] Cash	Yes (*)	No	No	No	Yes	Yes	No	No	No	No

(*) See explanation under FXOTheoretical section of limitations on analytical formulas for Lookback.

2.9 FX Option – Compound

Settlement type	PricerFXO									
	Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
[Late] Physical	Yes	No	No	No	No	No	Yes	Yes	Yes	Yes
[Late] Cash	Yes	No	No	No	No	No	Yes	Yes	Yes	Yes

2.10 FX Option – VolFwd

Settlement type	PricerFXO									
	Theoretical	VannaVolga	Heston	Market	Theoretical MonteCarlo	LocalVolatility MonteCarlo	Theoretical FiniteDifference	LocalVolatility FiniteDifference	Heston FiniteDifference	LocalStochasticVolatility FiniteDifference
[Late] Cash	Yes	No	No	No	No	No	No	No	No	No

2.11 Variance Swap – FX

Swap Type	Conditional Volatility Type	With Cap	Fwd Start	Settlement type	PricerFXO			
					Theoretical	CarrLee	Theoretical MonteCarlo	LocalVolatility MonteCarlo
Variance	None	No	No, Yes	Cash, Compo	Yes	Yes	Yes	Yes
Variance	None	No	No, Yes	3rd Ccy Quanto	Yes	No	Yes	No
Variance	None	Yes	No	Cash, Compo	Yes	Yes	Yes	Yes
Variance	None	Yes	No	3rd Ccy Quanto	Yes	No	Yes	No
Variance	None	Yes	Yes	Cash, Compo	Yes	No	Yes	Yes
Variance	None	Yes	Yes	3rd Ccy Quanto	Yes	No	Yes	No
Variance	Ups, Downs,Corr	No	No, Yes	Cash, Compo	Yes	No	Yes	Yes
Variance	Ups, Downs,Corr	No	No, Yes	3rd Ccy Quanto	Yes	No	Yes	No
Variance	Ups, Downs,Corr	Yes	No, Yes	Cash, Compo	No	No	Yes	Yes
Variance	Ups, Downs,Corr	Yes	No, Yes	3rd Ccy Quanto	No	No	Yes	No
Volatility	None	No	No	Cash, Compo	Yes	Yes	Yes	Yes
Volatility	None	No	No	3rd Ccy Quanto	Yes	No	Yes	No
Volatility	None	No	Yes	Cash, Compo	Yes	No	Yes	Yes
Volatility	None	No	Yes	3rd Ccy Quanto	Yes	No	Yes	No
Volatility	None	Yes	No	Cash, Compo	Yes	Yes	Yes	Yes
Volatility	None	Yes	No	3rd Ccy Quanto	Yes	No	Yes	No
Volatility	None	Yes	Yes	Cash, Compo	Yes	No	Yes	Yes
Volatility	None	Yes	Yes	3rd Ccy Quanto	Yes	No	Yes	No
Volatility	Ups, Downs,Corr	No, Yes	No, Yes	Cash, Compo	No	No	Yes	Yes
Volatility	Ups, Downs,Corr	No, Yes	No, Yes	3rd Ccy Quanto	No	No	Yes	No

2.12 Variance Option – FX

Swap Type		Fwd Start	Settlement type	PricerFXO			
				Theoretical	CarrLee	Theoretical MonteCarlo	LocalVolatility MonteCarlo
Variance		**No**	**Cash, Compo**	**Yes**	**Yes**	**Yes**	**Yes**
Variance		**No**	**3rd Ccy Quanto**	**Yes**	**No**	**Yes**	**No**
Variance		**Yes**	**Cash, Compo**	**Yes**	**No**	**Yes**	**Yes**
Variance		**Yes**	**3rd Ccy Quanto**	**Yes**	**No**	**Yes**	**No**
Volatility		**No**	**Cash, Compo**	**Yes**	**Yes**	**Yes**	**Yes**
Volatility		**No**	**3rd Ccy Quanto**	**Yes**	**No**	**Yes**	**No**
Volatility		**Yes**	**Cash, Compo**	**Yes**	**No**	**Yes**	**Yes**
Volatility		**Yes**	**3rd Ccy Quanto**	**Yes**	**No**	**Yes**	**No**

Section 3. PricerFXOTheoretical

PricerFXOTheoretical calculates prices for FX Options under the Black & Scholes (B&S) model.

The B&S formulation used is the one typically applied to FX Options:

$$\frac{dX_t}{X_t} = \mu(t)dt + \sigma(t)dW_t$$

where:

- X_t is the observed exchange rate at time t.
- $\mu(t)$ is a curve of known-in-advance, instantaneous, drifts for the returns of X_t.
- $\sigma(t)$ is a curve of known-in-advance, instantaneous, volatilities for the returns of X_t.

Please observe that under B&S preliminary assumptions, both $\mu(t)$ and $\sigma(t)$ curves are known in advance, andsupposed to stay as predicted as time goes by.

Once the B&S theory is developed following Risk Neutral arguments, and adding the assumption of continuous dynamic hedging, and zero transaction costs, it is shown that the original equation will produce the same results asthe more convenient:

$$\frac{dX_t}{X_t} = \frac{\frac{\partial F(t_0,t)}{\partial t}}{F(t_0,t)} dt + \sigma(t)dW_t$$

where $F(t', t")$ is the price for a forward delivering at $t"$ observed at t'. However, and again, the new drift term $(\partial F(t0,t) / \partial t) / F(t0,t)$ is assumed to stay stable as time goes by.

Regarding $\sigma(t)$, the natural situation is to have a calibrated curve of expiry-linked average volatilities. i.e. a curve of volatilities $\sigma(t_0,t)$ valid at time t_0 to be plugged into the classical B&S vanilla formula when calculating the price of a vanilla expiring at t.

In that case, the desired formulation can still be recovered as:

$$\frac{dX_t}{X_t} = \frac{\frac{\partial F(t_0,t)}{\partial t}}{F(t_0,t)} dt + \sqrt{\overline{\sigma}(t_0,t)^2 + 2 \cdot \overline{\sigma}(t_0,t) \cdot (t - t_0) \cdot \frac{\partial \overline{\sigma}(t_0,t)}{\partial t}}\, dW_t$$

always keeping the assumption that $\sqrt{\overline{\sigma}(t_0,t)^2 + 2 \cdot \overline{\sigma}(t_0,t) \cdot (t - t_0) \cdot \frac{\partial \overline{\sigma}(t_0,t)}{\partial t}}$ stays stable as time goes by.

All the indicated assumptions tend to disqualify the B&S model as a mark to market scheme. However, the B&Smodel is nevertheless widely used in the FX Option market for many reasons, including:

- Analytical tractability: the simplicity of the model allows the use of closed-form solutions (or good proxies)for many types of options, which allows:
  - Fast valuation and analysis of large portfolios.

  - Stable results and greeks.
- Simplicity: even though the model underlying assumptions are rather simplistic, their flaws are well understood by practitioners. Some portfolio managers prefer to track their risks using a model with knownflaws and rich reports over a more complex one with not so easy to understand assumptions, and limited reporting capabilities.
- Standardized prices: the theory is widespread, and almost every trading institution has access to a B&S price calculator. Announcing the B&S price of an operation being dealed, allows both parts to confirm thatthey are having the same understanding of such operation.

The main reason why this pricer has been developed, though, is to serve as basis for Vanna Volga valuation. Quick, homogeneous, accessible and, most importantly, stable and smooth Black & Scholes formulas are critical in order to ensure that Vanna Volga prices are themselves consistent and stable.

3.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
TV_ATM_TYPE	**Domain** A curve of ATM volatilities needs to be extracted from volatility surfaces to be input into theoretical pricing. Two main methods are supported: - Market Concordant: ATM volatility will respect definition indicated (or implied) by the respective FX surface conventions. - Forward: ATM Forward volatility will be used. i.e., the volatility for a strike equals to the respective expiry's forward will be used.	Market Concordant
TV_USE_FLAT_TERM_STRUCTURE	**Boolean** If set to true, the term structure of volatilities and rates is not taken into account at the time of generating parameters for the B&S model. Rather than that, the ones picked for maturity are used. Useful for validating the implementation against text book formulas. Not recommended for calculation of sensitivities or risk reports, as the product would only show sensitivities to the near-expiry pillars.	false

Pricing Parameter	Description	Typical Value
TV_ASIAN_ARITH_PROXY	**Domain** No exact Black & Scholes formula exists for valuation of Asian options when the average is of Arithmetic type. Good proxy formulas exist, making unnecessary using Monte Carlo to complete valuation. This parameter allows to choose among the different proxies: - Log Normal - Shifted Log Normal	Log Normal
TV_CONDITIONAL_VOL_SWAP_MODEL	**Domain** Black & Scholes Formulas for Conditional Volatility Swaps are not available. Since Calypso pricer configuration scheme doesn't allow to configure Conditional Volatility Swaps separately, this escape parameter has been prepared. A Conditional Volatility Swaps valued with this pricer, will be implicitly valued using the model hinted by this parameter. Possible values are (*): - TheoreticalMonteCarlo / TVMC: valuate Conditional Volatility Swaps using PricerFXOTheoreticalMonteCarlo. - Best: valuate Conditional Volatility Swaps using the best model available for it (which is PricerFXOTheoreticalMonteCarlo nowadays). (*) Filling this pricer parameter is mandatory in the case that ConditionalVolatility Swaps are valuated with PricerFXOTheoretical. The intention is to ensure that the useris aware of the limitations of the FXOTheoretical model before escaping to alternative valuation.	Best

3.2 References

The following references are provided as a contrast reference. Actual implementation can vary slightly in order toobtain improved performance or accuracy.

European Vanilla

Haug, E.G. The Complete Guide to Option Pricing Formulas 2nd edition. New York: Mc-Graw Hill. 2006. p 7-9

European Binary

Haug, E.G. The Complete Guide to Option Pricing Formulas 2nd edition. New York: Mc-Graw Hill. 2006. p 174, 175

American Vanilla

Haug, E.G. The Complete Guide to Option Pricing Formulas 2nd edition. New York: Mc-Graw Hill. 2006. p 299-303

Full Time Simple Barrier, Full Time Simple Touch

Haug, E.G. The Complete Guide to Option Pricing Formulas 2nd edition. New York: Mc-Graw Hill. 2006. p 152-155

Full Time Double Barrier, Full Time Double Touch

Haug, E.G. The Complete Guide to Option Pricing Formulas 2nd edition. New York: Mc-Graw Hill. 2006. p 156-160

Partial Time Simple Barrier, Partial Time Simple Touch

Armstrong, G.F. Valuation *formulae for window barrier options*. Applied Mathematical Finance, Volume 8, Issue 4,2001. p197-208

Other Partial Barrier options

Analytical formulas are derived from the principles that can be found at

Doob, J. L. Heuristic Approach to the Kolmogorov-Smirnov Theorems. The Annals of Mathematical Statistics, Vol. 20, Issue 3 (Sep, 1949), p 393-403

Anderson, T. W. A Modification of the Sequential Probability Ratio Test to Reduce the Sample Size. The annals of Mathematical Statistics, Vol. 31, Issue 1 (Mar. 1960), p 165-167.

Asian

Haug, E.G. The Complete Guide to Option Pricing Formulas 2nd edition. New York: Mc-Graw Hill. 2006. p 182-202.

Turnbull, S. L. & Wakeman, L. M., A Quick Algorithm for Pricing European Average Options, Journal of Financial and Quantitative Finance, 26, 377-389. (1991).

Brigo, D., Mercurio, F., Rapisarda, F., Scotti, R. Approximated moment-matching dynamics for basket-options simulation, Quantitative Finance, Vol 3, N.3 (2003) pp. 173-183.

Skipper, M. & Buchen, P. A Valuation Formula for Multi-Asset, Multi-Period Binaries in a Black-Scholes Economy, The ANZIAM Journal, Vol. 50, Issue 4 (Apr. 2009), pp. 475-485.

Lookback

Haug, E.G. The Complete Guide to Option Pricing Formulas 2nd edition. New York: Mc-Graw Hill. 2006. p 141-148.

M. Broadie, P. Glasserman, and S. Kou. Connecting Discrete and Continuous Path Dependent Options. Finance and Stochastics, 3:55-82, 1999.

Skipper, M. & Buchen, P. A Valuation Formula for Multi-Asset, Multi-Period Binaries in a Black-Scholes Economy, The ANZIAM Journal, Vol. 50, Issue 4 (Apr. 2009), pp. 475-485.

Variance Swap, Volatility Swap

Clark, Iain J. Foreign exchange option pricing : a practitioner's guide. John Wiley & Sons Ltd (2011). p 214-224.

3.3 Implementation details

It is important to note that final implementation cannot be directly inferred from referenced textbook formulas.

The main reason for this, is that textbook formulas tend to ignore some specific details that should be accounted for in practice, like the fact that the spot price is valid for a particular date in the short future, but not for instantaneous exchange; or the fact that in some markets the day count for volatilities and interest rates are not compatible; or the fact that late settlement require for specific treatment, etc.

Introducing all those facts in a Black & Scholes valuation is a simple exercise, but it unavoidably makes the final implementation differ from what can be found in text books or papers.

Additionally, some of the formulas have been refined beyond what can be directly found in documentation sources, with the intention to incorporate as much as possible of the richness coming from the term structure of volatilities and interest rates.

The reason for this decision is that, even though direct reproducibility can help on the process of validating formulas, it is understood that consistency with other more elementary prices (discount factors, forwards, etc.) will be much more important in general.

3.4 Current Limitations

Barrier Formulas

The analytical B&S formulas used for full time, partial time and multi-period barriers, assume that the volatility andinterest rates are constant across their relevant sections. Speaking volatility, for full time barriers it's assumed thatthere is a unique volatility which is valid in the whole life of the option. For partial barriers, it's assumed that there is a volatility to apply between contract date and barrier start date, another volatility to apply between barrier startdate and barrier end date, and finally another volatility to apply for the period between barrier end date and expirydate. The "in section" volatilities, are implied from the volatilities that would be used to valuate vanillas expiring at each of the relevant dates.

This simplified solution is however the one preferred by practitioners and so, the most likely to match the TVresults of different institutions.

Barrier Formulas – Rebates – Negative discount rate

The B&S analytical formula for rebates (where an amount is paid immediately after hitting the barrier), is valid fora certain range of discount factors (discount factor to expiry date, that is). All discount factors below 1 (equivalentto zero or positive interest rate) are accepted, and the discount factors above 1 are accepted up to a certain level,

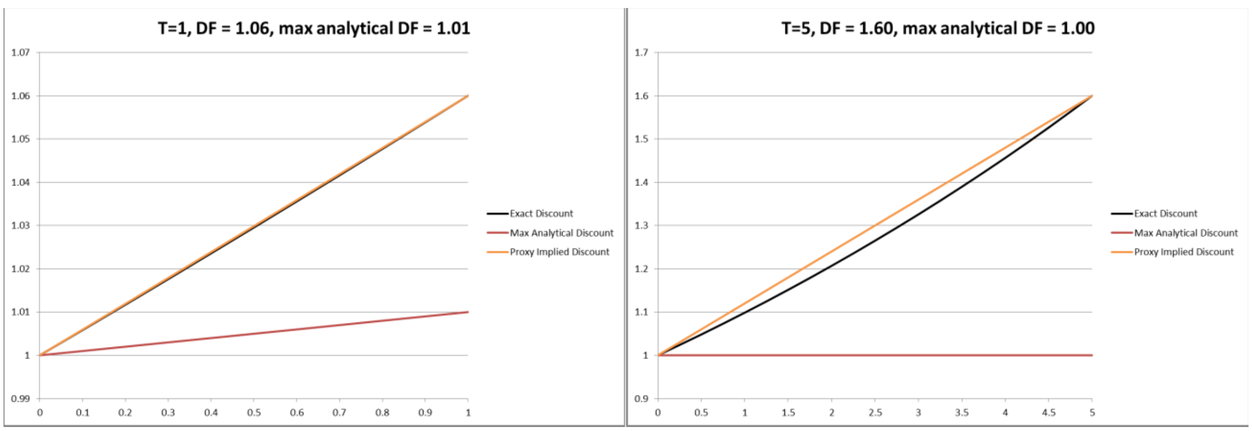
df_{max}, which depends in the rest of parameters being involved.

While negative interest rates are not the norm, they are observed in certain market circumstances. To adapt tothem, a proxy is used in the event that they appear at a level higher than df_{max}.

$$PV\ Rebate(B,s,f,\sigma,df) \approx PV\ Rebate(B,s,f,\sigma,df_{max}) + \frac{\partial PV\ Rebate(B,s,f,\sigma,x)}{\partial x}(df_{max}) \cdot (df - df_{max})$$

This is simply a first order Taylor expansion of the formula available for valid interest rates, against the discount factor df, centered at df_{max}. While a first order Taylor expansion is rather elementary, the particular variable of theexpansion was chosen among others (expansion could have been done against the interest rate, for instance) because it fared better as an approximation.

The quality of the approximation can be observed in the next two plots, which show the discount at every possible hit time implied by different criteria, and under different circumstances each plot.



It can be seen that the discount factor implied by the proxy tracks very closely the exact one, even in the very adverse circumstances of the second plot.

It can also be appreciated that the proxy has the effect of overestimating the discount factor to apply, and hence the price of the rebate, i.e. in an environment of negative interest rates, it produces results as if the discount interest rate was more negative than it really is.

In any case, since the effect of the proxy is to change the way that the discount factor is interpolated by the model between the current date and the observation date of the barrier.

Asian Formulas

It is not possible to produce exact B&S formulas for the valuation of Arithmetic Asian options. To fully account for all the aspects of Arithmetic Asian formulas, a dedicated Monte Carlo pricer would be necessary, thus incurring in ahigh performance cost.

Fortunately, very good proxy formulas exist, and two have been made available in Calypso:

- Log Normal proxy: also known as Turnbull-Wakeman approximation, or two moment matching. It’s very fast and stable, but it can miss a little detail when the non-volatile part of the average is relatively big.

- Shifted Log Normal proxy: the arithmetic average, instead of being approximated by as a rough log normal variable, is approximated as a constant plus a log normal variable. The extra constant term allowsto better accommodate the non-volatile part of the average. The third moment of the arithmetic average is used to optimize the constant term.

In practice, Log Normal proxy should suffice. However, if some extra accuracy is desired, Shifted Log Normal canbe used instead. Please take into account that Shifted Log Normal is computationally more expensive, as a numerical quadrature can be necessary in order to calculate the price for certain Asian configurations.

Lookback Formulas

Exact lookback formulas exist, but they require the application of multi-dimensional normal distribution quadratures. The number of dimensions matches the number of fixings participating in the lookback.

Unfortunately, multi-normal quadratures are known and stable up to dimension 3, only. For that reason, the B&S valuation of lookback options is:

- Exact, if the number of lookback fixings is not bigger than 3.
- Approximated, if the number of lookback fixings exceeds 3.

When the number of fixings exceeds 3, the lookback option is approximated by the formulas for a continuous

lookback, adapting it by application of the usual Broadie-Glassermann-Kou approximation.

The B-G-K correction works assuming that the discrete fixings are regularly distributed between first and last fixing dates. For that reason, the pricing should not be relied upon when applied on a lookback with more than 3 fixings if those are highly irregularly distributed.

Volatility Swap, Variance Option, Volatility Option Formulas

While exact formulas exist for Variance Swaps, an approximation is necessary for Volatility Swap, Variance Options and Variance Swaps.

All approximation are started by estimating exactly a few moments of the realized variance, applying standard Black & Scholes theory.

In the case of a Volatility Swap, the first and second moment of realized variance are computed. Then, since $Realized\_volatility = \sqrt{Realized\_variance}$, it is relatively easy to find approximations of moments of the realized volatility using a Taylor expansion. For instance, denoting X the realized variance, and Y the realized volatility:

$$E[Y] = E[\sqrt{X}] \approx E\left[\sqrt{X_0} + \frac{1}{1!}\frac{1}{2\sqrt{X_0}}(X - X_0) + \frac{1}{2!}\frac{-1}{4X_0\sqrt{X_0}}(X - X_0)^2 + \cdots\right] = E\left[\sqrt{X_0} + \frac{1}{2\sqrt{X_0}}(X - X_0) - \frac{1}{8X_0\sqrt{X_0}}(X - X_0)^2 + \cdots\right]$$
$$= \sqrt{X_0} + \frac{1}{2\sqrt{X_0}}(E[X] - X_0) - \frac{1}{8X_0\sqrt{X_0}}E[X^2] + \frac{1}{4\sqrt{X_0}}E[X] - \frac{1}{8}\sqrt{X_0} + \cdots$$

Therefore, the first moment of Y, $E[Y]$ can been approximated using the few first moments of X: $E[X]$, $E[X2]$, … This approximated value of $E[Y]$ can be used directly to compute the value of a Volatility Swap.

For Variance Options, or Volatility Options, it is necessary to add further assumptions. Typically up to 3 moments of realized variance are computed. Then, a shifted lognormal variable is fitted to the moments calculated for the realized variance, in a fashion similar to what is done for arithmetic Asian options. An integration is then carried using this approximate distribution of realized variance to estimate the value of an option on realized variance, or realized volatility.

The number of realized variance moments used in each case is as follows:

	Realized Variance Moments
Volatility Swap	2
Variance Option	3
Volatility Option	3

Section 4. PricerFXOMarket

The Black & Scholes model departs from a bunch of over-simplifying assumptions and so a very easily tractable model is derived.

However, in practice the price of European vanilla options across a range of maturities and strikes is derived from a volatility surface which, in turn, is tabulated to the market. Assuming that such information is available, by elementary replication arguments, the price of any derivative that can be decomposed as a combination of vanillas, can be easily calculated as the combination of the prices of the components.

Furthermore, certain options, like European binaries and Self-Quanto options can be replicated to a great degree with a portfolio of vanillas. Indeed, a replicating portfolio as close to those payouts as desired can be generated. Those theoretically perfect-replicating portfolios are not possible in practice (due to the unrealistic assumption of zero transaction costs and infinite liquidity), but they are useful to provide a reference mid price.

In practice, the collection of options that can be valuated by this mechanism coincides with that of European options, i.e., options whose payout depends only on the fixing of the currency rate at a specific date in the future.

Furthermore, adding the assumption of no correlation between interest rates and spot rate, European options with late delivery can be valuated with this methodology as well.

PricerFXOMarket is a pricer that uses those arguments whenever possible or reasonable to obtain the price of an option.

In practice, the collection of options covered reduces to Vanillas, European binaries, and all European barriers (i.e., barriers at EXPIRY).

Calypso implementation also includes some other products below this pricer for practical reasons. See Section 2 for detailed information on coverage. Please observe that this is naturally the pricer to choose for valuation of European vanilla options.

4.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
MKT_SELF_QUANTO_MODEL	**Domain** It applies to self quanto options only. Calypso doesn't allow to spare configuration of self-quanto options independently from other more standard option types. This parameter allows to escape the hard choice making possible the use of more than one model under the same pricer. Possible values are: - Theoretical / TV: valuate self-quanto options using PricerFXOTheoretical.	Best

Pricing Parameter	Description	Typical Value
	- VannaVolga / VV: valuate self-quantooptions using PricerFXOVannaVolga. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate self- quanto options using PricerFXOLocalVolatilityMonteCarlo. - Market / Mkt: valuate self-quanto options using PricerFXOMarket. - Best: uses the best available model, depending on the option type. Vanna Volga will be used for European vanillas, and Barriers / Digitals / Digitals with Barrier, when the barrierduration is EXPIRY.	
MKT_THIRD_CCY_QUANTO_MODEL	**Domain** Quanto options, even when they are European, can't be valuated with the sole premises defining the FXOMarket pricer. Since Calypso pricer configuration scheme doesn't allow to configure quanto options separately from other more FXOMarket friendly types, this escape parameter has been prepared. The quanto options will be valued using the model hinted by this parameter. Possible values are (*): - Theoretical / TV: valuate quanto options using PricerFXOTheoretical. - Best: valuate quanto options using the best model available for it (which is PricerFXOTheoretical nowadays). (*) Even though there is only one possible value, filling this pricer parameter is mandatoryin the case that quanto options are going to be valuated. The intention is to ensure that the user is aware of the limitations of the FXOMarket model before escaping to alternativeTV valuation.	Best
MKT_ACCRUAL_FADER_MODEL	**Domain** Fader Accrual options, can't be valuated with the sole premises defining the FXOMarket pricer. Since Calypso pricer configuration scheme doesn't allow to configure Fader Accrual options	Best

Pricing Parameter	Description	Typical Value
	separately from other more FXOMarket friendly types, this escape parameter has been prepared. The Fader Accrual options will be valued using the model hinted by this parameter. Possible values are (*): - Theoretical / TV: valuate Fader Accrual options using PricerFXOTheoretical. - VannaVolga / VV: valuate Fader Accrual options using PricerFXOVannaVolga. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate Fader Accrual options using PricerFXOLocalVolatilityMonteCarlo. - LocalVolatilityFiniteDifference / Local Vol Finite Difference / LVFD: valuate Fader Accrual options using PricerFXOLocalVolatilityFiniteDifference - LocalStochasticVolatilityFiniteDifference / Local Stoch Vol Finite Difference / LSVFD: valuate Fader Accrual options using PricerFXOLocalStochasticVolatilityFiniteDifference. - Best: valuate Fader Accrual using the best model available for it (which is PricerFXOLocalStochasticVolatilityFiniteDifference nowadays). (*) Filling this pricer parameter is mandatory in the case that Fader Accrual options are going to be valuated. The intention is to ensure that the user is aware of the limitations of the FXOMarket model before escaping to alternative valuation.	
MKT_ASIAN_CASH_IN_PRIMARY_MODEL	**Domain** Asian options with cash settlement in primary currency, when they are of type Average Rate or Double Average, can't be valuated with the sole premises defining the FXOMarket pricer. Since Calypso pricer configuration scheme doesn't allow to configure such options separately from other more FXOMarket friendly types, this escape parameter has been prepared. The primary-cash-Asian options will be valued using the model hinted by this parameter. Possible values are (*):	Best

Pricing Parameter	Description	Typical Value
	- Theoretical / TV: valuate them using PricerFXOTheoretical. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate them using PricerFXOLocalVolatilityMonteCarlo. - Best: valuate them using the best model available for it (which is PricerFXOLocalVolatilityMonteCarlo nowadays). (*) Filling this pricer parameter is mandatory in the case that Fader Accrual options are going to be valuated. The intention is to ensure that the user is aware of the limitations of the FXOMarket model before escaping to alternative valuation.	

Note: please be aware that, in the case that you configure PricerFXOMarket in such a way that PricerFXOTheoretical or PricerFXOVannaVolga are used implicitly in certain scenarios, you will need to complete your configuration with the parameters appropriate for PricerFXOTheoretical and/or PricerFXOVannaVolga.

Additionally, the calculation of the pricer measure TV will always require an invocation to the Theoretical model, and at that point the available PricerFXOTheoretical configuration parameters will be used to decide how to calculate it.

4.2 References

Self-quanto

Mercurio, F. Pricing and static replication *of FX quanto options*. Banca IMI Internal Report available http://www.fabiomercurio.it/FwdStartQuanto.pdf, 2003.

4.3 Implementation Details

American Vanilla Volatilities

Some cases of American Vanillas are accepted under this pricer. The valuation is carried using the B&S paradigm,with volatilities picked at the strike level for each step in the trinomial tree used for calculation.

Or in other words, if $\sigma(T, K)$ is a function giving the interpolated volatility for an European vanilla with strike K expiring at T, and the American vanilla being valuated has as strike $\tilde{K}$ then the Black & Scholes equation is parameterized as

$$\frac{dX_t}{X_t} = \frac{\frac{\partial F(t_0,t)}{\partial t}}{F(t_0,t)} dt + \sqrt{\frac{\partial \bar{\sigma}(t,\tilde{K})^2}{\partial t}} dW_t$$

Some cases of Fwd Start options are accepted under this pricer. The valuation is carried using the B&S paradigm, using two different volatilities for the relevant dates of the option:

- $\sigma START = \sigma(tSTART, F(t0, tSTART))$
- $\sigma EXPIRY = \sigma(T, \tilde{K})$ where $\tilde{K}$ is the forecast of the final strike, and is estimated as follows:

  $x\,\%\,atmf$: $x/100 \cdot * F(t_0,T)$

  $x\,\%\,atms$: $x/100 \cdot * F(t_0,t_{START})$

  $x\,\%\,otm$:$(1+\varphi * x/100) \cdot * F(t_0,t_{START})$, φ being +1 for a call, -1 for a put

  $pips$: $F(t_0,t_{START}) + x/10000$

With these two volatilities, an implied average volatility for the period (t_{START},T) can be calculated and used for the valuation. Please observe that even though the volatility choice could be object of debate, it at least ensures convergence to the respective vanilla as option approaches the fixing event at t_{START}.

Asian Volatilities

Some cases of Asian options are accepted under this pricer. The valuation is carried using the B&S paradigm, usingadjusted volatilities for the relevant dates of the option. The selection of volatilities is done in such a way that convergence to the valuation of European Vanillas and Fwd Start options is guaranteed.

The main criteria is the one applied over Asian subtypes *Average Rate* and *Geom Avg Rate*.

For an *Average Rate* subtype, with strike K, where the rate is given as $S = \sum^n w_i S_i$, with schedule dates

$t_1, \dots, t_k, t_{k+1}, \dots, t_n$, respective schedule weights $w_1, \dots, w_k, w_{k+1}, \dots, w_n$, known past fixings $S_1, \dots, S_k$, and forwards forfuture fixings $F_{k+1}, \dots, F_n$, then for every future date t_i, with $i \in \{k, \dots, n\}$, its respective volatility is chosen as:

$$\sigma_i = \bar{\sigma}(t_{START}, K_i)$$

$$K_i = F_i \cdot \left(1 + \sqrt{e^{\bar{\sigma}(t_i,F_i)^2 \cdot \bar{t}_i} - 1} \cdot \bar{x}\right)$$

$$\bar{x} = \frac{K - \left(\sum_{i=1}^{k} w_i S_i + \sum_{i=k+1}^{n} w_i F_i\right)}{\sum_{i=k+1}^{n} w_i F_i \sqrt{e^{\bar{\sigma}(t_i,F_i)^2 \cdot \bar{t}_i} - 1}}$$

where $\bar{t}_i$ is the volatility time given by the day count underlying the volatility surface.

The rationale behind the given criteria is to search for a common number of future ATM forward deviations, $\bar{x}$, that would make the resulting future Average Rate match the strike K.

Since such a search is difficult to carry in an exact way, as it requires iteration, the lognormal variates have been replaced by their normal proxy:

$$F \cdot e^{-\frac{\sigma^2 t}{2}+\sigma\sqrt{t}\cdot x} \sim F \cdot \left(1 + \sqrt{e^{\sigma^2 \cdot t} - 1} \cdot x\right)$$

Care is taken while applying the rule to avoid problems arising from possible negative values of the strikes K_i.

In the case of *Geom Avg Rate*, where the rate is given as $S=\prod_{i=1}^{n} S_i^{w_i}$, same rationale can be applied, but in this case it is not necessary to use any approximation, as a direct formula can be derived:

$$\sigma_i = \overline{\sigma}(t_i, K_i)$$

$$K_i = F_i \cdot e^{-\frac{\overline{\sigma}(t_i, F_i)^2 \cdot \bar{t}_i}{2} + \overline{\sigma}(t_i, F_i) \cdot \sqrt{\bar{t}_i} \cdot \bar{x}}$$

$$\bar{x} = \frac{log\left(\frac{K}{\prod_{i=1}^{k} S_i{}^{w_i} \cdot \prod_{i=k+1}^{n} F_i{}^{w_i} \cdot e^{-\frac{1}{2}\sum_{i=k+1}^{n} w_i \overline{\sigma}(t_i, F_i)^2 \cdot \bar{t}_i}}\right)}{\sum_{i=k+1}^{n} w_i \, \overline{\sigma}(t_i, F_i) \cdot \sqrt{\bar{t}_i \cdot}}$$

For the *Average Strike, Geom Avg Strike, DARO, Geom DARO* cases, assuming that the strike schedule dates are before (or at least not after) any date in the rate schedule, then the smile adjustment will leverage on the ideathat, once the strike is fully known, the remaining option will be either an European vanilla (*Average Strike, Geom Avg Strike* cases), or an *Average Rate* or *Geom Avg Rate* Asian (*DARO*, *Geom DARO* cases, respectively), for which a smile criteria is already known.

Therefore many future scenarios are drawn at the last date of the strike schedule. Those scenarios are simulated using ATM Forward volatilities for the fixings in the strike schedule. Then, for each of those scenarios, a valuationof the remaining European Vanilla / Rate Asian option is performed applying their known smile criteria.

To improve the performance of such a simulation of future scenarios, the future strike is approximated by the sametechnique chosen for Theoretical asian valuation. Thus the future strike is reduced to a unique random variate correlated with the spot at the last strike date, and the simulation of future scenarios becomes a simulation of two random variates which, thanks to its reduced dimensionality, can be carried using highly accurate integration techniques.

In each future scenario, a future volatility surface is necessary. Such a future volatility surface is synthetized from the original one using the following formula:

$$\bar{\bar{\sigma}}(S, T, K) = \sqrt{\frac{\bar{\sigma}^2\left(T, F(t_0, t_S) \cdot \frac{K}{S}\right) \cdot \bar{T} - \bar{\sigma}^2\left(t_S, F(t_0, t_S)\right) \cdot \bar{t}_S}{\bar{T} - \bar{t}_S}}$$

where

- $\bar{\bar{\sigma}}(S,T,K)$ is the future volatility for a European Vanilla option expiring at *T*, with strike *K*, in an scenario with spot at last strike fixing date being *S*.
- $F(t_0,t_S)$ is the forward of the spot at last strike fixing date t_S as estimated from current market conditions.

- $\bar{t}_S$, T are the volatility times given by the daycount underlying the volatility surface, for both, the last strike fixing date and the expiry time of the considered vanilla.

Please observe that even though this future volatility methodology could be object of debate, it at least ensures convergence to the respective vanilla option as the last strike date tS is approached.

Care is taken to avoid problems arising from likely situations where the formula cannot be applied without incurring mathematical inconsistencies.

4.4 Current Limitations

An important aspect to take into account, is that this methodology relies on the interpolated volatility surface as the source of vanilla prices. For that reason, all limitations coming from the volatility surface interpolation will beinherited by this pricer. Or in other words, limitations in the volatility surface interpolation technology could poselimitations on this methodology.

Particularly, the valuation of self-quanto vanillas by replication, poses strong conditions on the volatility surface in order to obtain a finite result. While interpolation is not very important in this respect, extrapolation is critical. Mostconfigurations possible within Calypso FX volatility surfaces lead to flat extrapolation, which ensures finite replication. In other circumstances, it should be checked whether the extrapolation meets the necessary conditions.A profound study of the conditions ensuring a finite replication of self-quant options can be found at:

Lee, R., The moment formula for implied volatility at extreme strikes, Mathematical Finance 14(3): 469-480, 2004.

Calypso does not currently support 3rd ccy quanto options under this pricer.

Section 5. PricerFXOVannaVolga

PricerFXOVannaVolga implements pricing using the Vanna Volga adjustment.

Unlike assumed by the Black & Scholes model, the observed volatility is not constant, but it evolves with time. Andnot only that, there is not such a thing as a unique observed volatility, not even a unique volatility curve, but rather a volatility surface.

The Vanna Volga adjustment is a technique to incorporate the extra information coming from the volatility surfaceto the pricing of a derivative.

The Vanna Volga technique produces a volatility surface adapted price following next steps:

1. Assume that the observed Smile prices are corrections over their respective Black & Scholes prices.

2. Assume that the corrections try to account for the fact that the volatility is dynamic, rather than constant.

3. Reinterpret those corrections as the cost of statically hedge the dynamic volatility risks, like Vega, Vanna or Volga.

4. Apply the calibrated costs to a target derivative, taking into account the dynamic volatility risks (Vega, Vanna, Volga) of such a derivative.

That simplistic view of the task of pricing derivatives, has been extensively researched in literature. In particular,Castagna and Mercurio have published theoretical analysis of the method, while Wystup has produced initial hintson market practice at the time of applying the adjustment.

For that reason, the topic will not be further developed here.

The wide presence in literature is a reflection of the extensive usage of Vanna Volga for valuation of derivatives.The main reason for its success, is that the method departs from the Black & Scholes model as a starting point.That means that:

- Adoption is cheap for houses with existing Black & Scholes valuation methodologies.
- The method inherits the analytical tractability of the Black & Scholes model.
- The method inherits the unbeatable speed of the Black & Scholes model as well.

However, please observe that those are all conveniency reasons. It should be stressed that the Vanna Volga adjustment is an *adjustment*, and not a solid model. So jumps in valuation, negative valuations for options with positive payouts, and similar aberrations are perfectly possible from a theoretical standpoint. Fortunately, the nature of FX markets somehow mitigates those possibilities, but in any case, this valuation methodology should beadopted with care.

5.1 Pricing Parameters

The abundant literature helps to discover that, the loose theoretical foundation of Vanna Volga pricing, allows for many variations at the time of implementing it. On an attempt to cover several of the different flavors, the pricerFXOVannaVolga is highly configurable, allowing for several of the most usual variations to be applied.

Pricing Parameter	Description	Typical Value
VV_REFERENCE_DELTA	**Double**	25

Pricing Parameter	Description	Typical Value
	Delta of the hedge vanillas to be used. Expressed in % directly. A value strictly between 0 and 50 is expected.	
VV_REFERENCE_DELTA_TYPE	**Domain** The ATM and Delta conventions used when searching respective strikes of the hedge vanillas. - Market Concordant: the convention indicated (or implied) by the respective FX Volatility Surface is respected. - Symmetric: a special definition is used for ATM and Delta. The definition ensures symmetry when changing perspective around the underlying pair. - Symmetric Proxy: same as Symmetric, but the search of the strike is done in a relaxed way: a short, fixed, number of search iterations will be performed. This will improve performance and stability of Vanna Volga pricing, and the difference against Symmetric will be unnoticeable in practice.	Market Concordant
VV_WEIGHT_POLICY	**Domain** It indicates how to calculate the fraction of the Vanna Volga adjustment to apply over the B&S theoretical value. - Full / F: a weight of 1 (the total adjustment is applied). - Expected Life Fraction / ELF: the quotient between the expected life of the deal, and its maximum possible life. - Survival Probability / SP: the probability of the deal reaching its maximum possible life. This weight will always be lower than the Expected Life Fraction. - Fisher / Fi: the probability adjustment suggested in Section 2 of Fisher's paper Variations on the Vanna-Volga Adjustment. - User Input / UI / U: input by the user (see VV_INPUT_WEIGHT below)	Expected Life Fraction

Pricing Parameter	Description	Typical Value
	Please observe that Expected Life Fraction and Survival Probability will produce always 1 for European options, including options with EXPIRY barriers.	
VV_INPUT_WEIGHT	**Domain** The fraction of the adjustment to be applied over the B&S theoretical value, as input by the user. Expected in fraction form (more specifically, a number between 0 and 1). In effect only if VV_WEIGHT_POLICY is set to User Input.	0.5 The default if not specified is 1
VV_WEIGHT_USE_SYMMETRIC_PROB	**Boolean** It applies to options with FULL or PARTIAL barriers only. Whether a symmetric B&S valuation should be used when calculating the adjustment weight (see VV_WEIGHT_POLICY). If classical B&S valuation is used to calculate the weighting probabilities, the result would depend on the numeraire currency used in the B&S model. A symmetric numeraire allows to obtain a number that is approximately the average of what it would be obtained with the other two numeraires. It also helps to obtain similar results regardless of the quotation used for the currency pair (either straight or inverted).	true
VV_HEDGE_POLICY	**Domain** Criteria for allocating hedge instruments to hedged risks. In general, the synthetic policy is the theoretical to be applied to cancel all the relevant risks. It also preserves symmetry of the adjustment against quotations. The natural policy, even though keeping residual risks after the VVV hedge, could produce better mark to market for certain types of exotics.	Synthetic

Pricing Parameter	Description	Typical Value
	There are two possible criterias right now: - Synthetic: the vega, vanna, volga of the position are all simultaneously matched by a portfolio conveniently composed from the VVV hedge options. - Natural: - Vega is matched with using a weighted straddle with same vega. - Vanna of the position is matched using a weighted Risk Reversal with same vanna. - Volga of the position is matched using a weighted Butterfly with same Volga.	
VV_HEDGE_EXPIRY	**Domain** It applies to options with FULL or PARTIAL barriers only. Criteria for the picking the expiry of the hedge portfolio. Please take into account that the hedge portfolio will carry the effect of the smile, and so it will place the sensitivity to smile from the volatility surface. The criteria to use when choosing the expiry of the hedge portfolio. - Barrier End Date (Especially indicated when using VV_SPARE_TERMINAL_ADJUSTMENT) - Expiry Date - Expected Exit Date. Please take into account that the Expected Exit Date will change with evolution of time and market.	Barrier End Date
VV_SPARE_TERMINAL_ADJUSTMENT	**Domain** It applies to options with FULL or PARTIAL barriers only. If activated, the option is assumed to be composed of two parts: Option = [αTPO] + [Option – αTPO] TPO stands for Terminal PayOut, and is the payout that will be made available in case the barrier condition is met before Expiry.	true

Pricing Parameter	Description	Typical Value
	α is the amount of TPOs which is considered, and is calculated as the probability of effectively reaching the maturity of the option. The final price is finally calculated as: αRef(TPO) + VV(Option – αTPO) Ref(PTO) is the price of PTO using the terminal reference model (see VV_TERMINAL_REF_MODEL). VV(Option – αTPO) is the price of the remainder deal Option – αTPO using Vanna Volga as configured by the other parameters. This technique ensures that consistency with vanillas are respected in every situation. It's best applied when VV_TERMINAL_REF_MODEL is set to Market.	
VV_TERMINAL_REF_MODEL	Domain It applies to options with FULL or PARTIAL barriers only. The model to use for valuating the terminal payout when applying the VV_SPARE_TERMINAL_ADJUSTMENT technique. - Market / Mkt - VannaVolga / VV	Market
VV_CONSISTENCY_ENFORCEME NT	**Domain** Since Vanna Volga is not a model, it can produce inconsistent results in many ways (like negative premiums for vanillas). Several checks can be put in place to correct the situation up to a certain extent. Those checks accept the following profiles: - None: no correction is done. Useful for study of the adjustment properties. - Basic: constrains that can be derived from the price of the terminal payout are applied (premium must be non negative, and lower than the vanilla price, for instance).	Basic

Pricing Parameter	Description	Typical Value
	- Advanced: a stronger attempt to keep several consistency rules is done. Please take into account that this level produces marginal results, and has a hit in performance for certain barrier types, as all the boundary payouts has to be calculated as well. See the Implementation Details to a brief summary of the boundaries, and their enabling in the different profiles.	
VV_5_POINTS_ADJUSTMENT *EXPERIMENTAL* *USE FOR STUDY PURPOSES ONLY*	**Boolean** It applies an extended version of the Vanna Volga technique, which accepts 5 hedge instruments, instead of the classical three. The extension has been studied and justified from a theoretical perspective, but its performance in real usage scenarios is still under study.	false
VV_REFERENCE_EXTRA_DELTA *EXPERIMENTAL* *USE FOR STUDY PURPOSES ONLY*	Delta of the two extra hedge vanillas to be used. Expressed in % directly. A value strictly between 0 and 50 is expected. It must also be different from VV_REFERENCE_DELTA.	10
VV_EXPIRY_BARRIER_MODEL	**Domain** It applies to options with EXPIRY barriers only. Even the usage of Vanna Volga for FULL and PARTIAL barriers is natural, EXPIRY barriers are best calculated using the PricerFXOMarket. This parameter allows to choose a different model than Vanna Volga to be applied for EXPIRY barriers. Possible values are: - Market / Mkt: valuate EXPIRY barriers using PricerFXOMarket - VannaVolga / VV: valuate EXPIRY barriers using Vanna Volga methodology	Market
VV_THIRD_CCY_QUANTO_MODEL	**Domain**	Best

Pricing Parameter	Description	Typical Value
	Pricer FXOVannaVolga has not been extended to handle Quanto options yet. Since Calypso pricer configuration scheme doesn't allow to configure quanto options separately from other more FXOVannaVolga friendly types, this escape parameter has been prepared. The quanto options will be valued using the model hinted by this parameter. Possible values are (*): - Theoretical / TV: valuate quanto options using PricerFXOTheoretical. - Best: valuate quanto options using the best model available for it (which is PricerFXOTheoretical nowadays). (*) Even though there is only one possible value, filling this pricer parameter is mandatory in the case that quanto options are going to be valuated. The intention is to ensure that the user is aware of the limitations of the FXOVannaVolga model before escaping to alternative TV valuation. (**) EXPIRY_BARRIER_MODEL will be checked first. In the event of having an assignment different from Vanna Volga model, the assigned EXPIRY_BARRIER_MODEL will be used to the evaluation of quanto expiry barriers, regardless of the value carried by VV_THIRD_CCY_QUANTO_MODEL.	
VV_ACCRUAL_RESETTABLE_MODEL	**Domain** Resettable Accrual options, can't be valuated under the current the FXOVannaVolga pricer. Since Calypso pricer configuration scheme doesn't allow to configure Resettable Accrual options separately from other more FXOVannaVolga friendly types, this escape parameter has been prepared. The Resettable Accrual options will be valued using the model hinted by this parameter. Possible values are (*):	Best

Pricing Parameter	Description	Typical Value
	- Theoretical / TV: valuate Resettable Accrual options using PricerFXOTheoretical. - Market / Mkt: valuate Resettable Accrual options using PricerFXOMarket. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate Resettable Accrual options using - PricerFXOLocalVolatilityMonteCarlo. Best: valuate Resettable Accrual using the best model available for it (currently pointing to PricerFXOMarket). (*) Filling this pricer parameter is mandatory in the case that Resettable Accrual options are going to be valuated. The intention is to ensure that the user is aware of the limitations of the FXOVannaVolga model before escaping to alternative valuation.	

Note: please be aware that, in the case that you configure PricerFXOVannaVolga in such a way that PricerFXOTheoretical or PricerFXOMarket are used implicitly in certain scenarios, you will need to complete yourconfiguration with the parameters appropriate for PricerFXOTheoretical and/or PricerFXOMarket.

Particularly, the calculation of the TV price, which serves as a base for the Vanna Volga adjustment, will adhere tothe available PricerFXOTheoretical flags.

5.1.1 Configuration of Fisher's methodology

The methodology indicated in *T. Fisher, Variations on the Vanna-Volga Adjustment*, can be exactly replicated ifthe following configuration is set:

- VV_REFERENCE_DELTA = 25
- VV_HEDGE_POLICY = Synthetic
- VV_WEIGHT_USE_SYMMETRIC_PROB = true
- VV_WEIGHT_POLICY = Fisher
- VV_TERMINAL_REF_MODEL = VannaVolga
- VV_CONSISTENCY_ENFORCEMENT = Advanced
- VV_SPARE_TERMINAL_ADJUSTMENT = False
- VV_HEDGE_EXPIRY = Expiry / Barrier End Date (Fisher doesn't address this case)
- VV_5_POINTS_ADJUSTMENT = false

The remaining settings are either unrelated to cases covered by Fisher, not clearly indicated in the paper, or have no effect due to the previous settings.

While this configuration tracks Fisher's paper accurately, there is no guarantee of reproducing same prices observed in Bloomberg.

5.1.2 Logging key

The FXOVannaVolga pricer can dump some details of its internal calculations to the Calypso log, by adding VannaVolgaDetails category to the log configuration.

5.2 References

A. Castagna and F. Mercurio, Consistent Pricing of FX Options, (2006)

A. Castagna FX Options and smile risk, Wiley Finance (2010)

U. Wystup, The market price of one-touch options in foreign exchange markets, Derivatives Week Vol. 12, n 13 London (2003)

P. Carr, A. Hogan and A. Verma, Vanna-Volga Method For 1st Generation Exotics in FX, Bloomberg Financial Markets Commodities News (March 9, 2006)

F. Bossens et al, Vanna-Volga methods applied to FX derivatives: from Theory to Market Practice. Int. J. Theor. Appl. Finan. 13, 1293 (2010)

T. Fisher, Variations on the Vanna-Volga Adjustment, Bloomberg Financial Markets Commodities News (January 26, 2007)

5.3 Implementation Details

The Vanna Volga method is straightforward from an implementation point of view. However, there are many little choices that can impact the final results.

Most of those choices can be edited using the respective Pricing Parameter, but there are several of them which are still not available as an option. It is understood that the impact of those choices on the results is not very large, but since some of those choices cannot be considered “standard”, they are listed here.

Reference Delta Type

The definition of Delta and ATM at play when selecting the hedge vanillas, is configured via the parameter VV_REFERENCE_DELTA_TYPE. When such a parameter is set to *Market Concordant*, the convention implied by therespective FX volatility surface will be respected. Only exception will be when the convention itself changes with maturity. In such a case, an intermediate convention will be used between latest pillar with the short term convention, and first pillar with the long term convention. That will ensure that:

- Vanna Volga applied with maturity before latest short term convention term pillar uses exact marketconvention.
- Vanna Volga applied with maturity after first long term convention pillar uses exact market convention.
- Vanna Volga applied with maturity between latest short and first long pillars, produces results thattransition smoothly between the results seen in the other cases.

When VV_REFERENCE_DELTA_TYPE is set to either *Symmetric* or *Symmetric Proxy*, then a special definition of Delta and ATM is used instead. ATM is same as popular convention ATM Forward. The definition of a Delta, however, is not corresponding to any of the 4 usual conventions observed in the market. It is indeed following thisexact formula (for a Call delta):

$$\Delta(t_0, t, K) = N\left(\frac{\log\left(\frac{F(t_0, t)}{K}\right)}{\bar{\sigma}(t_0, t, K) \cdot \sqrt{t - t_0}}\right)$$

This definition of delta is not conventional, but it has certain advantages, like helping preserve symmetry when changing the way the currency pair is quoted, or not being impacted by discount factors, or being naturally compatible with the ATM strike being defined as the forward, since $\Delta(t_0,t,F(t_0,t))=0.5$ = 50%.

Please observe that the mapping $\bar{\sigma}(t_0,t,K)$ is the actual input internally available to the Vanna Volga pricer for its calculations. However, the Vanna Volga technique needs to obtain the strikes for some pre-configured definitions of ATM and Delta (typically ATM, 25 Delta Call and 25 Delta Put), and that cannot be directly got from the available $\bar{\sigma}(t_0,t,K)$ mapping.

The solution is logically to do a search on the possible strikes until one giving the desired $\Delta(t_0,t,K)$ is found.

When REFERENCE_DELTA_TYPE is set to *Market Concordant* or *Symmetric*, a full search to a tight tolerance will be performed.

The drawback of such a search, is that it can be expensive, as it would require to use the mapping $\bar{\sigma}(t_0,t,K)$ many times until convergence is assured. A classical way to reduce the cost, would be to increase the tolerance to the error when searching for the delta. But large tolerances could lead to unstable solutions, which in turn would translate into unstable prices and particularly unstable pricer measures.

As an alternative, when REFERENCE_DELTA_TYPE is set to *Symmetric Proxy*, the search is performed up to a short, fixed number of iterations. That ensures that the results of the iterative search are smooth against smooth changes in the market parameters.

With this later setting, the iteration does not converge exactly to the target symmetric delta. However, we have considered it to be an affordable deviation, as it is a well known feature of Vanna Volga the fact that it is self consistent for vanilla prices and thus, small deviations in the hedge vanilla strikes should produce negligible deviations on the final adjustment.

Consistency rules

Since Vanna Volga adjustment does not constitute a sound model, it gives room for inconsistencies and anomalies on the adjusted prices.

It is usual practice to enforce some practical consistency by checking whether the usual price is in bounds, and rectifying it in case it is not.

However, as more and more types of derivatives are added to the spectrum of Vanna Volga, and more and more consistency constrains among them appear, soon the exercise becomes a puzzle with difficult solution, and the attempts to keep consistency obfuscate the rationale behind the pricer.

Next table lists some of the consistency rules that are applicable for Digitals and Barriers, and from which level of VV_CONSISTENCY_ENFORCEMENT pricing parameter they are available, or under which conditions in case they are not directly controlled by that parameter.

For the case of Digital With Barrier, same logic applies than for Barrier.

Barrier duration	Relationship	Minimum VV_CONSISTENCY_ENFORCEMENT level	Notes
Any	0<=NTU<=Cash	Basic	
Any	0<=NTD<=Cash	Basic	
Any	0<=DNT<=Cash	Basic	
Any	NTU=0 at the barrier	#N/A	Met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction or SurvivalProbability
Any	NTD=0 at the barrier	#N/A	Met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction or SurvivalProbability
Any	DNT=0 at any barrier	#N/A	Met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction or SurvivalProbability
Any	UO(oo)=Cash	None	It occurs naturally, no need to enforce it
Any	DO(0)=Cash	None	It occurs naturally, no need to enforce it
Any	DNT(0,oo)=Cash	None	It occurs naturally, no need to enforce it
Any	NTU+OTU=Cash	None	It occurs naturally, but problems could appear if any of the terms has to be either floored or capped
Any	NTD+OTD=Cash	None	It occurs naturally, but problems could appear if any of the terms has to be either floored or capped
Any	DNT+DOT=Cash	None	It occurs naturally, but problems could appear if any of the terms has to be either floored or capped
Any	DOT(L,H)<=OTD(L)+OTU(H)	Advanced	DOT is reduced if necessary
Any	DNT(L,H)<=NTD(L)	Advanced	DNT is reduced if necessary

Barrier duration	Relationship	Minimum VV_CONSISTENCY_ENFORCEMENT level	Notes
Any	DNT(L,H)<=NTU(H)	Advanced	DNT is reduced if necessary
Any	OTNT_UI(L,H) = NTD(L)-DNT(L,H)	None	
Any	OTNT_DI(L,H) = NTD(H)-DNT(L,H)	None	
Any	UO Call - UO Put = NTU EUR - K*NTU USD	None	It occurs naturally, but problems could appear if any of the terms has to be either floored or capped
Any	DO Call - DO Put = NTD EUR - K*NTD USD	None	It occurs naturally, but problems could appear if any of the terms has to be either floored or capped
Any	DKO Call - DKO Put = DNT EUR - K*DNT USD	None	It occurs naturally, but problems could appear if any of the terms has to be either floored or capped
Any	0<=UO<=Vanilla	Basic	
Any	0<=DO<=Vanilla	Basic	
Any	0<=DKO<=Vanilla	Basic	
Any	UO=0 at the barrier	#N/A	Met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction or SurvivalProbability
Any	DO=0 at the barrier	#N/A	Met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction or SurvivalProbability
Any	DKO=0 at any barrier	#N/A	Met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction or SurvivalProbability
Any	UO(oo,K)=Vanilla(K)	#N/A	Met if VV_SPARE_TERMINAL_ADJUSTMENT is set to true. It will match the Vanilla price calculated with the model indicated by VV_TERMINAL_REF_MODEL.

Barrier duration	Relationship	Minimum VV_CONSISTENCY_ENFORCEMENT level	Notes
Any	DO(0,K)=Vanilla(K)	#N/A	Met if VV_SPARE_TERMINAL_ADJUSTMENT is set to true. It will match the Vanilla price calculated with the model indicated by VV_TERMINAL_REF_MODEL.
Any	DKO(0,oo,K)=Vanilla(K)	#N/A	Met if VV_SPARE_TERMINAL_ADJUSTMENT is set to true. It will match the Vanilla price calculated with the model indicated by VV_TERMINAL_REF_MODEL.
Any	UO+UI=Vanilla	None	It will match the Vanilla price calculated with the model indicated by VV_TERMINAL_REF_MODEL.
Any	DO+DI=Vanilla	None	It will match the Vanilla price calculated with the model indicated by VV_TERMINAL_REF_MODEL.
Any	DKO+DKI=Vanilla	None	It will match the Vanilla price calculated with the model indicated by VV_TERMINAL_REF_MODEL.
Any	DKI(L,H,K) = UIDO(L,H,K) + UODI(L,H,K)	None	UIDO and UODI are rescaled if necessary
Any	UIDO(L,H,K)<=UI(H,K)	Advanced	UIDO is reduced if necessary
Any	UODI(L,H,K)<=DI(L,K)	Advanced	UODI is reduced if necessary
Any	DKO(L,H,K)<=DO(L,K)	Advanced	DKO is reduced if necessary
Any	DKO(L,H)<=UO(H,K)	Advanced	DKO is reduced if necessary
Any	KIKO_UI(L,H,K) = DO(L,K)-DKO(L,H,K)	None	
Any	KIKO_DI(L,H,K) = UO(H,K)-DKO(L,H,K)	None	

Barrier duration	Relationship	Minimum VV_CONSISTENCY_ENFORCEMENT level	Notes
Full	DNT_USD(L,H) = {DKO(L,H,K=L,Call) + DKO(L,H,K=H,Put)} / (H-L)	None	It occurs naturally, but problems could appear if any of the terms has to be either floored or capped
Full	DNT_EUR(L,H) = {H*DKO(L,H,K=L,Call) + L*DKO(L,H,K=H,Put)} / (H- L)	None	It occurs naturally, but problems could appear if any of the terms has to be either floored or capped
Point	OTU(H)=NTD(H)	#N/A	Met if VV_WEIGHT_POLICY is set to Full. Also met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction and VV_HEDGE_EXPIRY is set to BarrierEndDate.
Point	OTD(L)=NTU(L)	#N/A	Met if VV_WEIGHT_POLICY is set to Full. Also met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction and VV_HEDGE_EXPIRY is set to BarrierEndDate.
Point	DOT(L,H)=OTD(L)+OTU(H)	#N/A	Met if VV_WEIGHT_POLICY is set to Full. Also met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction and VV_HEDGE_EXPIRY is set to BarrierEndDate.
Point	OTU, OTD, NTD, NTU could be much better approximated by call-spread.	#N/A	Promoted if VV_WEIGHT_POLICY is set to Full. Also promoted if VV_WEIGHT_POLICY is set to ExpectedLifeFraction and VV_HEDGE_EXPIRY is set to BarrierEndDate.
Point	UI(H,K)=DO(H,K)	#N/A	Met if VV_WEIGHT_POLICY is set to Full. Also met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction, VV_HEDGE_EXPIRY is set to BarrierEndDate and VV_SPARE_TERMINAL_ADJUSTMENT is set to true.
Point	DI(L,K)=UO(L,K)	#N/A	Met if VV_WEIGHT_POLICY is set to Full. Also met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction, VV_HEDGE_EXPIRY is set to BarrierEndDate and

Barrier duration	Relationship	Minimum VV_CONSISTENCY_ENFORCEMENT level	Notes
			VV_SPARE_TERMINAL_ADJUSTMENT is set to true.
Point	DKI(L,H,K)=DI(L,K)+UI(H, K)	#N/A	Met if VV_WEIGHT_POLICY is set to Full. Also met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction, VV_HEDGE_EXPIRY is set to BarrierEndDate and VV_SPARE_TERMINAL_ADJUSTMENT is set to true.
Expiry	Could be exactly replicated as a combination of vanillas and binaries	#N/A	Met if VV_WEIGHT_POLICY is set to ExpectedLifeFraction or Full. Expiry barriers can also be controlled by the pricing parameter EXPIRY_BARRIER_MODEL.

5.4 Current Limitations

Even though the implementation supports valuation of self-quanto options, it has not been extended to supportvaluation of 3rd ccy quanto options.

Please read the Implementation Details for a deeper understanding of some other choices made in the currentimplementation.

Section 6. PricerFXOHeston

PricerFXOHeston implements pricing using the Heston model.

More specifically, the Heston model obeys the following Stochastic Differential Equation:

$$\frac{dX_t}{X_t} = \mu(t)dt + \sqrt{v_t}dW_t^X$$

$$dv_t = \kappa(t)(\theta(t) - v_t)dt + \eta(t)\sqrt{v_t}dW_t^v$$

$$\langle dW_t^X, dW_t^v \rangle = \rho(t)dt$$

where:

- Xt is the observed exchange rate at time t.
- vt is the instantaneous variance at time t.
- $\mu(t)$ is a curve of known-in-advance, instantaneous, drifts for the returns of X_t.
- $\theta(t)$ is the mean variance. It is the value that v tends towards at time t.
- $\kappa(t)$ is the mean reversion. It is the speed at which v tends towards (reverts to) θ at time t.
- $\eta(t)$ is the volatility of variance at time t.
- $\rho(t)$ is the instantaneous correlation between the two driving Brownian motions W_{tX} and W_{tv}.

It can be appreciated that the equation driving the exchange rate is similar to that of the Black & Scholes model, only that the volatility, instead of being known in advance, is stochastic as well, and governed by the second equation.

This characteristic makes the Heston model richer than Black & Scholes, and better a priori, since it accounts for the real world fact that the instantaneous volatility is not known in advance, and will change in unpredictable ways. Indeed, and unlike the Black & Scholes model, the Heston model can produce skew and smile in the volatility surface.

Such fact does not imply that the Heston model is perfect. It can be argued that the exchange rate dynamics generated by the Heston equation are not realistic if compared with what it can be observed in the FX market.

However, the Heston model is still very popular due to its analytical tractability. Be it realistic or not, the Heston model is one of the few stochastic volatility models for which semi-analytical formulas for vanilla pricing are known. Therefore, when dealing with vanillas, which occurs naturally and intensively when calibrating its parameters, it is a very fast model.

The pricer FXOHeston pricer implements semi-analytical pricing of FX vanillas and, in general, FX European options, using the Heston model. It is not expected that the model is used in production for the valuation of such products, since a much simpler and faster alternative can be found in the pricer FXOMarket. However, since Heston dynamics are part of more sophisticated pricers (FXOHestonFiniteDifference, FXOLocalStochasticVolatilityFiniteDifference), FXOHeston can be used as a benchmark when validating such pricers.

6.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
H_REFERENCE_DELTA	**Double** Delta of the wing vanillas to be used for calibration. Expressed in % directly. A value strictly between 0 and 50 is expected.	25
H_MEAN_REVERSION_1Y	**Double** The mean reversion speed to use foroptions expiring in one year time. For options with expiry date differentto one year, the mean reversion is calculated as: $\kappa_{TY} = \kappa_{1Y} / \mathrm{T}$ where T is the time to expiry measuredin ACT/365 years.	1

6.2 Logging key

If the logging key *Analytics.FXO.Heston* is active, a trace with details about the calibration results will appearin the log.

6.3 References

Heston Formula for Vanillas

L. Andersen and V. Piterbarg. Interest Rate Modelling. Voumel I: Foundations and Vanilla Models. Atlantic FinancialPress, London New York. p 315-353

Vanilla Price to Volatility Methodology

Jäckel, P., Let's Be Rational. Wilmott, 2015: 40–53. doi: 10.1002/wilm.10395

6.4 Implementation Details

The Heston model is first simplified to a version with constant Heston parameters:

$$\frac{dX_t}{X_t} = \mu(t)dt + \sqrt{v_t}dW_t^X$$

$$dv_t = \kappa(\theta - v_t)dt + \eta\sqrt{v_t}dW_t^v$$

$$\langle dW_t^X, dW_t^v \rangle = \rho dt$$

The Heston parameters to consider are therefore reduced to: $v_0,\theta,\kappa,\eta,\rho$, where v_0 is the variance at the initial instant, t_0. These five parameters need to be known at the time of calculating the price of the option under consideration.

The strategy to discover them, which is executed right before every pricing, is as follows:

- κ is calculated from the user input under pricer parameter H_MEAN_REVERSION_1Y. $\kappa={}_{\kappa 1YT}$, where T is the time to expiry of the option in ACT/365 years.
- $v0$ is linked to the mean variance by the simple relation: $v0\equiv\theta$.
- The remaining parameters θ,η,ρ are calibrated so that the model reproduces the prices of the following three vanilla options:

  ATM option with same expiry as the option being priced.

  Δ Delta call option with same expiry as the option being priced. Δ given by pricer parameter H_REFERENCE_DELTA.

  Δ Delta put option with same expiry as the option being priced. Δ given by pricer parameter H_REFERENCE_DELTA.

Since the last three parameters are adjusted to precisely three vanilla prices, usually the calibration routine will find an almost exact solution, where all three prices are matched to very small difference.

Please observe that:

1. Different calibrated parameters will be found for options with different expiry.

2. The calibration is done on the fly as part of the pricing exercise. There is no place in the system to pre-specify the Heston parameters.

Once the Heston parameters have been found, the pricing of the European option being considered can be easily achieved by decomposing it into vanillas. Vanillas can be priced directly with the Heston formula, binaries can be priced using a call spread of vanillas, self-quanto options can be valued using 4.2's decomposition, etc.

6.5 Current limitations

This model is constrained to work on single exchange rate, European options. That includes vanillas, binaries and European barriers. It also includes valuation of self-quanto vanillas, but not any type of 3rd ccy quanto options.

Section 7. PricerFXOTheoreticalMonteCarlo

PricerFXOTheoreticalMonteCarlo calculates prices for FX Options under the Black & Scholes (B&S) model via a Monte Carlo integration.

The modelling assumptions are exactly the same as those exposed in Section 3 for PricerFXOTheoretical, and won't be indicated here. Therefore, both pricers should produce very similar results, except that the numerical methodology used for the calculation is different.

Since PricerFXOTheoretical already covers most of the existing products with an analytical, or semi-analytical formula, and such formula is much faster and precise than the Monte Carlo algorithm, this pricer is not expected to be used in practice.

However, it can be effectively used for validation purposes: a Monte Carlo algorithm, while slow and imprecise, is very simple to implement correctly, and therefore is a good tool to verify that analytical formulas, which are more involved to derive, have been derived correctly. It also helps to have a measure of the error incurred when the analytical formula is a proxy, taking the Monte Carlo result (using high precision) as a benchmark.

7.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
TV_ATM_TYPE	**Domain** A curve of ATM volatilities needs to be extracted from volatility surfaces to be input into theoretical pricing. Two main methods are supported: - Market Concordant: ATM volatility will respect definition indicated (or implied) by the respective FX surface conventions. - Forward: ATM Forward volatility will be used. i.e., the volatility for a strike equals to the respective expiry's forward will be used.	Market Concordant
TV_USE_FLAT_TERM_STRUCTURE	**Boolean** If set to true, the term structure of volatilities and rates is not taken into account at the time of generating parameters for the B&S model. Rather than that, the ones picked for maturity are used. Useful for validating the implementation against text book formulas. Not recommended for calculation of sensitivities or risk reports, as the product would only show sensitivities to the near-expiry pillars.	False

Pricing Parameter	Description	Typical Value
TV_MC_ITERATIONS	**Integer** The number of simulations to be used by the Monte Carlo engine. It has to an integer between 1 and 16777215, and it has to be a possible result of the formula $2n - 1$. For instance: 32767, 65535, 262143, etc. If those requisites are not met, the model will internally adjust the iterations to the next higher valid number. The higher the number of iterations, the higher the precision of the results, and the slower the calculation.	32767
TV_MC_PAYOFF_SMOOTHING_PV	**Double** The grade of smoothing to apply to the PV measures (PV, NPV, etc.). It has to be a positive number. 0 will mean no smoothing applied, while a positive number d, will mean d standard deviations.	0
TV_MC_PAYOFF_SMOOTHING_SNS	**Double** The grade of smoothing to apply to the sensitivity measures (DELTA, GAMMA, VEGA, THETA, etc.). It has to be a positive number. 0 will mean no smoothing applied, while a positive number d, will mean d standard deviations.	0.125

7.2 References

Jäckel, P. Monte Carlo Methods in Finance. John Wiley & Sons, Ltd, Chichester (2002)

7.3 Implementation Details

The implementation of a Black & Scholes Monte Carlo simulation is a standard procedure, and there is little choiceleft for a particular implementation.

Random number generation

The random number generation is performed via a Sobol sequence, as suggested in chapter 8 of *Monte CarloMethods in Finance* book.

Brownian Path generation

This particular implementation uses the Brownian Bridge technique as suggested in Section 10.8.3 of *Monte CarloMethods in Finance* book.

Payoff Smoothing

This model calculates the necessary sensitivities via finite differences. This technique is a general one, that is known to work well for smooth functions.

Therefore, it will work relatively well in a Monte Carlo setup, as long as the Monte Carlo PV function is smooth against perturbations of the relevant market parameters. In turn, the Monte Carlo PV function will be smooth as long as all the intermediate steps in the chain to produce the PV react smoothly to their inputs.

Since the same random number generation seed, and the same number of iterations are used consistently along all the PV calculations involved in finite differences algorithm, it can be assumed that the set of scenarios will react smoothly to perturbations of Market Data.

However, the payoff simulated in each scenario will not necessarily react smoothly to smooth changes in the generated scenarios. That's the typical case of a binary option, with its payout being:

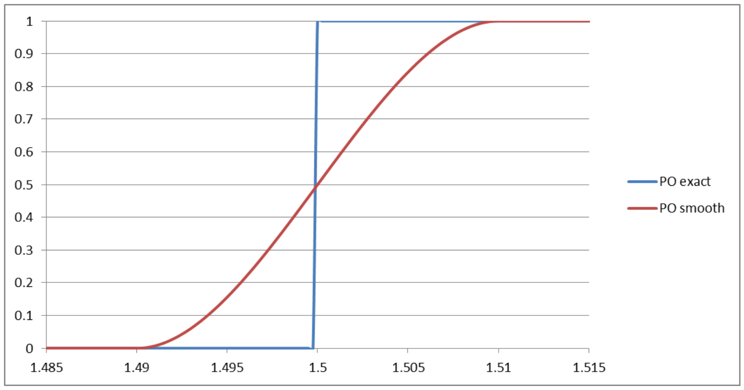
$$PO(S_T) = \begin{cases} 1 \; if \; S_T > K \\ 0 \; if \; S_T \leq K \end{cases}$$

Assuming that a draw *i* of S_T from the Monte Carlo simulation produces S_{Ti} very close to K (let's assume S_{Ti}=K), and that delta is computed by bumping the spot by a small amount $h_S>0$, the contribution of the draw *i* to the delta is:

$$Delta_i \approx \frac{PO\left(S_T^i + h_S \frac{dS_T^i}{dS_0}\right) - PO(S_T^i)}{h_S} = \frac{1-0}{h_S} = \frac{1}{h_S}$$

Please observe that this contribution is inversely proportional to the finite difference step size h_S, so when h_S is small enough, a random contribution like this can blow up the average $Delta$ result, and when h_S is big enough for that effect to be less dramatic, then the cost is having a poor $Delta$ approximation as a consequence of too big step size h_S (indeed, as h_S grows larger, more draws will fall into this pattern).

The solution that has been applied to this problem, has consisted of replacing the exact payoff by a smooth approximation:



When using the smooth approximation, the contribution to $Delta$ estimation out of draw *i* becomes:

$$Delta_i = \frac{PO\_smooth(S_T^i + h_S) - PO\_smoth(S_T^i)}{h_S} \approx \left(\frac{dPO_{smooth}}{dS_T} | S_T^i\right) \frac{dS_T^i}{dS_0}$$

It can be observed that this contribution is not so heavily depending on the size of the step h_S anymore.

This technique is therefore useful to stabilize Monte Carlo sensitivities estimated via finite differences, but it has thedisadvantage of replacing the true payout by a proxy which will, naturally, induce a bias in the PV and the estimated sensitivities themselves. That is why configuration flags have been made available to control its application and, indeed, different parameters apply to PV and to sensitivities: TV_MC_PAYOFF_SMOOTHING_PVand TV_MC_PAYOFF_SMOOTHING_SNS.

When one of those parameters is set to 0, then the technique is not applied and direct payout is used. When it'sset to a number d>0, all edges in the payout being evaluated are smoothed using the technique. Particularly, everyintermediate payout element with the formulation:

$$\begin{cases} f_{UP}(x) & if\ x > x^* \\ c & if\ x = x^* \\ f_{DOWN}(x) & if\ x < x^* \end{cases}$$

is replaced by this smoothed formulation:

$$\begin{cases} f_{UP}(x) & if\ x \geq x^* + s \\ p\left(\frac{x - x^*}{s}\right) \cdot f_{UP}(x) + \left(1 - p\left(\frac{x - x^*}{s}\right)\right) \cdot f_{DOWN}(x) & if\ x^* - s \leq x \leq x^* + s \\ f_{DOWN}(x) & if\ x \leq x^* - s \end{cases}$$

where:

$$p(y) = \frac{1}{2} + \frac{3}{4}y - \frac{1}{4}y^3$$

$$s = \frac{d}{2} \cdot standard\_deviation(x - x^*)$$

$standard\_deviation(x{-}x_*)$ meaning the standard deviation of the random quantity $x{-}x_*$ estimated out of the Monte Carlo sample.

Section 8. PricerFXOLocalVolatilityMonteCarlo

PricerFXOLocalVolatilityMonteCarlo calculates prices for FX Options under the Local Volatility (LV) model via a Monte Carlo integration.

The formulation used is the one typically applied to FX Options:

$$\frac{dX_t}{X_t} = \mu(t)dt + \sigma_{LV}(t, X_t)dW_t$$

where:

- X_t is the observed exchange rate at time t.
- $\mu(t)$ is a curve of known-in-advance, instantaneous, drifts for the returns of X_t.
- $\sigma_{LV}(t,X_t)$ is a surface of known-in-advance, instantaneous, local volatilities for the returns of X_t.

Please observe that under LV preliminary assumptions, both $\mu(t)$ curve and $\sigma_{LV}(t, X_t)$ surface are known in advance,and supposed to stay as predicted as time goes by.

Once the theory is developed following Risk Neutral arguments, and adding the assumption of continuous dynamichedging, and zero transaction costs, it is shown that the original equation will produce the same results as the more convenient:

$$\frac{dX_t}{X_t} = \tilde{\mu}(t)dt + \sigma_{LV}(t, X_t)dW_t$$

where:

- $\tilde{\mu}(t) = \frac{\frac{\partial F(t_0,t)}{\partial t}}{F(t_0,t)}$
- $F(t', t")$ is the price for a forward delivering at $t"$ observed at t'.

However, and again, the new drift term $\tilde{\mu}(t)$ is assumed to stay stable as time goes by.

Regarding $\sigma_{LV}(t,X_t)$, the theory from Dupire establishes a direct relationship between the price of vanillas of the underlying pair, and the local volatility. This relationship, combined with the direct relationship between vanilla prices and their volatilities, leads to this direct formula:

$$\sigma_{LV}(t, X_t) = \dot{\sigma}_{LV}\left(t, \frac{X_t}{F(t_0, t)}\right)$$

$$\dot{\sigma}_{LV}(t, Z) = \sqrt{\frac{\dot{\sigma}^2(t, Z) + 2 \cdot \dot{\sigma}(t, Z) \cdot (t - t_0) \cdot \frac{\partial \dot{\sigma}}{\partial t}(t, Z)}{\left(1 + Z \cdot \sqrt{t - t_0} \cdot d_1 \cdot \frac{\partial \dot{\sigma}}{\partial Z}(t, Z)\right)^2 + \dot{\sigma}(t, Z) \cdot Z^2 \cdot (t - t_0) \cdot \left(\frac{\partial^2 \dot{\sigma}}{\partial Z^2}(t, Z) - d_1 \cdot \left(\frac{\partial \dot{\sigma}}{\partial Z}(t, Z)\right)^2\right)}}$$

where:

- $\tilde{\sigma}(t,Z) = \bar{\sigma}(t_0, F(t_0,t), t, Z \cdot F(t_0,t))$
- $\bar{\sigma}(t_0, X_0, T, K)$ is the volatility for a European vanilla option on the exchange rate X, expiring at T and struck at K, being observed at instant t_0 and with the exchange rate at t_0 being X_0.

It can be observed that, in the particular case that $\bar{\sigma}(t_0,X_0,T,K_1)=\bar{\sigma}(t_0,X_0,T,K_2)$, for all T,K_1,K_2, then the formula for $\sigma_{LV}(t,X_t)$ becomes:

$$\sigma_{LV}(t,X_t) = \sqrt{\bar{\sigma}^2(t_0,X_0,t,*) + 2 \cdot \bar{\sigma}(t_0,X_0,t,*) \cdot (t-t_0) \cdot \frac{\partial \bar{\sigma}}{\partial t}}$$

Clearly, this coincides with the formulation of the B&S model as described in section PricerFXOTheoretical. So theB&S model can be seen as a particular case of the Local Volatility model when the volatility surface is flat with respect to moneyness, i.e., presenting flat smile.

Alternatively, the Local Volatility model can be seen as the simplest generalization of the B&S model that matchesthe price of vanillas in the presence of smile. The simplest in the sense that it is the only one generalization whichcan be expressed with only one state variable, X_t.

All these facts amount to the main features of the Local Volatility model:

- It is a natural extension of classic B&S model.
- It is fully compatible with the market price of European options, since it matches the prices of marketEuropean vanillas.
- Unlike Vanna Volga adjustment, it is a full blown model, with a solid theory behind.
- It is the simpler model to do so.

However, there are some caveats:

- In general, it requires numerical integration (like Monte Carlo), which leads to relatively slow and unstableimplementations, as compared to other models accepting analytical formulas.
- The dynamics implied by the model, are not entirely consistent with those observed in the market for FXrates.

Particularly for Calypso FX Options, this model can be a good choice for certain flavors of Asian, and it is the bestavailable choice for Lookback and some variations of Accrual.

8.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
LV_MC_MAX_STEP_DAYS	**Double**	14

Pricing Parameter	Description	Typical Value
	The maximum size, in days, of a time step in the discretization scheme. The lower the number of days, the higher the accuracy of the results, and the slower the calculation.	
LV_MC_ITERATIONS	**Integer** The number of simulations to be used by the Monte Carlo engine. It has to an integer between 1 and 16777215, and it has to be a possible result of the formula $2n - 1$. For instance: 32767, 65535, 262143, etc. If those requisites are not met, the model will internally adjust the iterations to the next higher valid number. The higher the number of iterations, the higher the precision of the results, and the slower the calculation.	32767
LV_MC_PAYOFF_SMOOTHING_PV	**Double** The grade of smoothing to apply to the PV measures (PV, NPV, etc.). It has to be a positive number. 0 will mean no smoothing applied, while a positive number d, will mean d standard deviations.	0
LV_MC_PAYOFF_SMOOTHING_SNS	**Double** The grade of smoothing to apply to the sensitivity measures (DELTA, GAMMA, VEGA, THETA, etc.). It has to be a positive number. 0 will mean no smoothing applied, while a positive number d, will mean d standard deviations.	0.125
LV_MC_TV_MODEL	**Domain** The model to be used for computation of TV. Two main methods are supported:	Theoretical

Pricing Parameter	Description	Typical Value
	- Theoretical: use same model as FXOTheoretical pricer. - TheoreticalMonteCarlo: use same model as FXOTheoreticalMonteCarlo pricer.	

Note: please be aware that, in the case that you configure PricerFXOLocalVolatilityMonteCarlo in such a way thateither PricerFXOTheoretical or PricerFXOTheoreticalMonteCarlo are used implicitly in certain scenarios, you will need to complete your configuration with the parameters appropriate for PricerFXOTheoretical and/or PricerFXOTheoreticalMonteCarlo.

8.2 References

Jäckel, P. Monte Carlo Methods in Finance. John Wiley & Sons, Ltd, Chichester (2002)

Dupire, B. Pricing with a smile, Risk, 7(1):18-20 (1994)

Gatheral, J. The volatility surface: a practitioner's guide, John Wiley & Sons (2006)

8.3 Implementation Details

Stochastic Differential Equation

The SDE used in practice, is the result of transforming the original one by applying Itô's Lemma:

$$dx_t = -\frac{1}{2}{\sigma_{LV}}^2(t, F(t_0,t) \cdot e^{x_t})\,dt + \sigma_{LV}(t, F(t_0,t) \cdot e^{x_t})\,dW_t = -\frac{1}{2}\dot{\sigma}_{LV}^2(t, e^{x_t})\,dt + \dot{\sigma}_{LV}(t, e^{x_t})\,dW_t$$

where:

- $x_t = \log \frac{X_t}{F(t_0,t)}$

This alternative, equivalent SDE has two main advantages for its practical application:

- The validity range for x_t is unbounded, and thus there is no need to apply extra checks on the simulatedvalues.
- The main component of the drift term disappears, and the volatility term appears in a way which isnaturally compatible with the intermediate formula for local volatilities.
-

Discretization

The necessary discretization of the SDE to be processed via a Monte Carlo engine, follows the classical, direct,Euler scheme:

$$x_{t_{i+1}} = x_{t_i} - \frac{1}{2}\dot{\sigma}_{LV}^2\left(t_i, e^{x_{t_i}}\right) \cdot (t_{i+1} - t_i) + \dot{\sigma}_{LV}\left(t_i, e^{x_{t_i}}\right) \cdot (W_{i+1} - W_i)$$

where the simulation instants t_i are chosen in such a way than the error due to discretization is small enough (the closer the instants to each other, the smaller the error due to discretization).

The Euler scheme typically requires a finely grained discretization to converge. That is so because the scheme fails to account for the fact that $\dot{\sigma}_{LV2}(t,e_{xt})$ will change between t_i and t_{i+1} as both its arguments t and x_t change (t due to the passage of time, and x_t due to the underlying Brownian motion driving it).

This requirement of a finely grained discretization, imposes an important overhead on the Monte Carlo method, as many discretization steps will be needed in order to their size small enough to keep the discretization error in check.

However, it is a known, practical fact that, when a $\dot{\sigma}_{LV}$ comes from a usual FX volatility surface, then:

- The speed of change of $\dot{\sigma}_{LV}$ with respect to t will slow down as t is further in the future.
- The speed of change of $\dot{\sigma}_{LV}$ with respect to x_t will slow down as t is further in the future.

For that reason, it seems practical to force smaller steps when t is near origin, and allow larger steps when t is further in the future. To implement that idea, this technique has been applied in the construction of the simulated instants:

- There is a *start step size*, currently stablished as a sixth of a day.
- There is a *maximum step size*, controlled externally by end user (via parameter LV_MC_MAX_STEP_DAYS).
- There is a constant ratio c currently stablished as 34.
- All simulation step sizes (the step size is the difference $\Delta t_i = t_{i+1} - t_i$) are built using this formula:

$$\Delta t_i \approx \min\left(\max\left(start\_step\_size \;\; , \; c \cdot \; (t_i - t_0)\right) \; , \; \text{maximum\_step\_size}\right)$$

In practice, some minor adjustments to the above technique are applied to ensure that the simulation instants imposed by the payoff being priced are included in the collection; however the maximum step size will never beexceeded.

Local Volatility Discretization

The formula for calculation of local volatilities is of instantaneous nature:

$$\dot{\sigma}_{LV}(t,Z) = \sqrt{\frac{\dot{\bar{\sigma}}^2(t,Z) + 2 \cdot \dot{\bar{\sigma}}(t,Z) \cdot (t - t_0) \cdot \frac{\partial \dot{\bar{\sigma}}}{\partial t}(t,Z)}{\left(1 + Z \cdot \sqrt{t - t_0} \cdot d_1 \cdot \frac{\partial \dot{\bar{\sigma}}}{\partial Z}(t,Z)\right)^2 + \dot{\bar{\sigma}}(t,Z) \cdot Z^2 \cdot (t - t_0) \cdot \left(\frac{\partial^2 \dot{\bar{\sigma}}}{\partial Z^2}(t,Z) - d_1 \cdot \left(\frac{\partial \dot{\bar{\sigma}}}{\partial Z}(t,Z)\right)^2\right)}}$$

i.e., this is the formula for the local volatility to apply in a jump from t to $t + \delta$ as $\delta \to 0$. However, after discretization, the local volatilities are needed for sizeable steps: from t_i to $t_{i+1} = t_i + \Delta t_i$, with $\Delta t_i > 0$. The aboveformula is therefore slightly adapted to work in this context:

$$\dot{\sigma}_{LV}\left(t_i \to t_{i+1}, Z\right) = \sqrt{\frac{\frac{\dot{\bar{\sigma}}^2(t_{i+1}, Z)\cdot(t_{i+1}-t_0) - \dot{\bar{\sigma}}^2(t_i, Z)\cdot(t_i - t_0)}{t_{i+1}-t_i}}{\left(\left(1 + Z\cdot\sqrt{t_{i+1}-t_0}\cdot d_1\cdot\frac{\partial\dot{\bar{\sigma}}}{\partial Z}(t_{i+1}, Z)\right)^2 + \dot{\bar{\sigma}}(t_{i+1}, Z)\cdot Z^2\cdot(t_{i+1}-t_0)\cdot\left(\frac{\partial^2\dot{\bar{\sigma}}}{\partial Z^2}(t_{i+1}, Z) - d_1\cdot\left(\frac{\partial\dot{\bar{\sigma}}}{\partial Z}(t_{i+1}, Z)\right)^2\right)\right)}}$$

Essentially, the expression originally in the numerator $\dot{\bar{\sigma}}^2(t, Z) + 2\cdot\dot{\bar{\sigma}}(t, Z)\cdot(t-t_0)\cdot\frac{\partial\dot{\bar{\sigma}}}{\partial t}(t, Z)$, which indeed is the same as $\frac{d\left[\dot{\bar{\sigma}}^2(t,Z)\cdot(t-t_0)\right]}{dt}$, has been replaced by the finite difference approximation of the later expression evaluated between t_i and t_{i+1}. This ensures that the volatility applied in the step is exact in the case when the FX volatility surface is flat with respect to moneyness.

Then the expression in the denominator has been evaluated at t_{i+1}(other possible "sensible" choices would have been evaluating it at t_i or indeed a t of the form $h\cdot t_i+(1-h)\cdot t_{i+1}$, after choosing an h in $[0,1]$). The choice of t_{i+1} was made in an attempt to preserve the smile information of the ending point, which typically is the one more important to reproduce for pricing.

Local Volatility Sampling

While the above formula for $\dot{\sigma}_{LV}(t_i \to t_{i+1}, Z)$ can be applied directly, doing so in practice would make the Monte Carlo simulation very slow. This is mainly due to the direct dependency on the market FX volatility surface $\bar{\sigma}$, and its derivatives, which typically are relatively slow. Just assuming that the calculation takes 1 millisecond, a simulation using 30000 monte carlo simulations and 50 discretization steps, would need 25 minutes to run.

To solve that problem, it is critical to ensure that the cost of calculating $\dot{\sigma}_{LV}(t_i \to t_{i+1}, Z)$ is minimal. The solution applied has consisted of sampling the discretized volatility $\dot{\sigma}_{LV}(t_i \to t_{i+1}, Z)$ in a few selected $Z_{ij}, j\ in\ \{1,\ldots,n_i\}$, yielding n_i points: $Z_{ij} \to \dot{\sigma}_{LV}(t_i \to t_{i+1}, Z_{ij})$, and then interpolating this reduced collection of points using a cubic spline for every Z needed by the simulation. The cost of invoking the cubic spline at every iteration is not only predictable, but very small.

The sampling points $\{Z_{ij}\}$ are selected so that they include classical FX volatility points (ATM, 25 Delta, 10 Delta, 1 Delta), plus a few other intermediate points to ensure that the interpolation stays close to the original formula.

Random number generation

The random number generation is performed the same way than for the pricer FXOTheoreticalMonteCarlo (see above).

Brownian Path generation

This generation of Brownian path variables W_i uses the same Brownian Bridge technique applied in the pricerFXOTheoreticalMonteCarlo (see above).

Payoff Smoothing

The payoffs being evaluated by this implementation, are smoothed using the same technique applied in the pricerFXOTheoreticalMonteCarlo (see above).

Section 9. PricerFXOTheoreticalFiniteDifference

PricerFXOTheoreticalFiniteDifference calculates prices for FX Options under the Black & Scholes (B&S) model via a Finite Difference integration.

The modelling assumptions are exactly the same as those exposed in Section 3 for PricerFXOTheoretical, and won't be indicated here. Therefore, both pricers should produce very similar results, except that the numerical methodology used for the calculation is different.

Since PricerFXOTheoretical already covers most of the existing products with an analytical, or semi-analytical formula, and such formula is much faster and precise than the Monte Carlo algorithm, this pricer is not expected to be used in practice.

However, it can be effectively used for validation purposes: the Finite Difference algorithm is the basis of some more sophisticated models available, and many of the valuation steps are shared among such sophisticated applications of Finite Difference and this one. The matching of this pricer against existing Theoretical analytical formulas should generate confidence on the general application of Finite Differences in FXO modeling.

9.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
TV_ATM_TYPE	**Domain** A curve of ATM volatilities needs to be extracted from volatility surfaces to be input into theoretical pricing. Two main methods are supported: - Market Concordant: ATM volatility will respect definition indicated (or implied) by the respective FX surface conventions. - Forward: ATM Forward volatility will be used. i.e., the volatility for a strike equals to the respective expiry's forward will be used.	Market Concordant
TV_USE_FLAT_TERM_STRUCTURE	**Boolean** If set to true, the term structure of volatilities and rates is not taken into account at the time of generating parameters for the B&S model. Rather than that, the ones picked for maturity are used. Useful for validating the implementation against text book formulas.	false

Pricing Parameter	Description	Typical Value
	Not recommended for calculation of sensitivities or risk reports, as the product would only show sensitivities to the near-expiry pillars.	
TV_FD_MAX_STEP_DAYS	**Double** The maximum size in days allowed for one Finite Difference time step. A smaller maximum size will increase the precision and accuracy of the results.	7
TV_FD_MIN_STEPS	**Integer** The minimum number of time steps to apply, regardless of the expiry of the option. A bigger number of minimum steps will increase the precision and accuracy for short term options.	75
TV_FD_DV2T_TO_DX2_RATIO	**Double** The average ratio of the size of the time step to the square of the size of the space step. More precisely: $ratio = \frac{\sigma^2 \Delta t}{(\Delta \log S)^2}$ This ratio is used to decide the size of the space step: $\Delta \log S = \sqrt{\frac{\sigma^2 \Delta t}{ratio}}$ A lower ratio will reduce the computational cost, and the precision in the space direction. A too high ratio will make oscillations in the solution more likely to appear.	0.5
TV_FD_THETA	**Double** The theta defining the Generalized Euler scheme to apply. Theta must be a number between 0 and 1.	0.5

Pricing Parameter	Description	Typical Value
	The following numbers will produce the named schemes: - 0: Explicit Euler - 0.5: Crank-Nicolson - 1: Implicit Euler	
TV_FD_MAX_THREADS	**Integer** The maximum number of threads that can be used by the model. The larger the number, the faster the pricing will be cleared. A number higher than the computer physical processor count will typically produce no marginal improvement. Parallelization occurs only at a per- measure level, so there will be no appreciable speed gain when applied to the calculation of PV only.	1

9.2 References

Tavella, D. & Randall, C. Pricing Financial Instruments: The Finite Difference method. John Wiley & Sons, Inc.(2000)

9.3 Implementation Details

As explained in Section 8, the Black & Scholes model can be seen as a particular case of the Local Volatility model.

Therefore, any technique valid to compute the price under Local Volatility model, can be easily reused to the sameeffect under Black & Scholes model.

Indeed, the implementation of Finite Difference solving is shared among both pricers: TheoreticalFiniteDifferenceand LocalVolatilityFiniteDifference.

The reader is kindly invited to refer to the Implementation Details detailed in Section 10.3, where explanation isprovided about the Finite Difference algorithm used for Local Volatility (and Black & Scholes) model.

Section 10. PricerFXOLocalVolatilityFiniteDifference

PricerFXOLocalVolatilityFiniteDifference calculates prices for FX Options under the Local Volatility model via a Finite Difference integration.

The modelling assumptions are exactly the same as those exposed in Section 8 for PricerFXOLocalVolatilityMonteCarlo, and will not be repeated here. Therefore, both pricers should produce similar results, except that the numerical methodology used for the calculation is different.

When compared to Monte Carlo integration, Finite Difference is superior in stability and speed, so it is recommended to choose LocalVolatilityFiniteDifference over LocalVolatilityMonteCarlo whenever it is desired to apply Local Volatility over a product that can be integrated by both methodologies.

10.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
TV_FD_MAX_STEP_DAYS	**Double** The maximum size in days allowed for one Finite Difference time step. A smaller maximum size will increase the precision and accuracy of the results.	7
TV_FD_MIN_STEPS	**Integer** The minimum number of time steps to apply, regardless of the expiry of the option. A bigger number of minimum steps will increase the precision and accuracy for short term options.	75
TV_FD_DV2T_TO_DX2_RATIO	**Double** The average ratio of the size of the time step to the square of the size of the space step. More precisely: $$ratio = \frac{\sigma^2 \Delta t}{(\Delta \log S)^2}$$ This ratio is used to decide the size of the space step: $$\Delta \log S = \sqrt{\frac{\sigma^2 \Delta t}{ratio}}$$ A lower ratio will reduce the computational cost, and the precision in the space direction. A too	0.5

Pricing Parameter	Description	Typical Value
	high ratio will make oscillations in the solution more likely to appear.	
TV_FD_THETA	**Double** The theta defining the Generalized Euler scheme to apply. Theta must be a number between 0 and 1. The following numbers will produce the named schemes: - 0: Explicit Euler - 0.5: Crank-Nicolson - 1: Implicit Euler	0.5
TV_FD_MAX_THREADS	**Integer** The maximum number of threads that can be used by the model. The larger the number, the faster the pricing will be cleared. A number higher than the computer physical processor count will typically produce no marginal improvement. Parallelization occurs only at a per- measure level, so there will be no appreciable speed gain when applied to the calculation of PV only.	1
LV_FD_TV_MODEL	**Domain** The model to be used for computation of TV. Two main methods are supported: - Theoretical: use same model as FXOTheoretical pricer. - TheoreticalFiniteDifference: use same model as FXOTheoreticalFiniteDifference pricer.	Theoretical

10.2 References

Tavella, D. & Randall, C. Pricing Financial Instruments: The Finite Difference method. John Wiley & Sons, Inc. (2000)

Andersen, B. G. & Piterbarg, V. V. Interest Rate Modeling, Volume I: Foundations and Vanila Models. 1st edition, Atlantic Financial Press, 2010.

Andersen, B. G. & Piterbarg, V. V. Interest Rate Modeling, Volume II: Term Structure Models. 1st edition, Atlantic Financial Press, 2010.

Dupire, B. Pricing with a smile, Risk, 7(1):18-20 (1994)

Gatheral, J. The volatility surface: a practitioner's guide, John Wiley & Sons (2006)

10.3 Implementation Details

The practice for implementation of Finite Difference valuation, is widely documented. Particularly the book indicatedin the 9.2 section, does a very detailed introduction and exposition of the basic principles.

In the following paragraphs, detail is provided in particular decisions taken while implementing the standard technique. They are presented as a sequence of steps, relationships and properties. However, no especial effort is done on justifying the particular decisions adopted, understanding that a quiet read of basic textbooks in thesubject will be much more effective to develop some intuition about the technique.

As indicated in Section 9.3, exploiting the fact that the Theoretical model can be seen as a particular case of the Local Volatility model, the implementation details explained here will be valid for both models at the same time: LocalVolatility Finite Difference, and Theoretical Finite Difference.

Partial Differential Equation

The PDE used in practice, is the Local Volatility PDE without any transformation:

$$-\frac{\partial V(t,X)}{\partial t} = \frac{1}{2}\,\sigma_{LV}(t,X)^2\,X^2\,\frac{\partial^2 V(t,X)}{\partial X^2} + \tilde{\mu}(t)\,X\,\frac{\partial V(t,X)}{\partial X} - r(t)\,V(t,X)$$

Eq. 1

where:

- $V(t,X)$ is the value of the derivative at time t if the spot at time t, X_t, took the value X.

A notation to represent special solutions of this PDEs will be introduced. If $PO(X)$, with $X\in[0,\infty[$ is used as an initial condition at t'' (i.e., imposing $V(t'',X)=PO(X),\forall X\in[0,\infty[$), then the particular solution of such PDE at t', with $t'<t''$, will be denoted as:

$$E_{t'\leftarrow t''}[PO(X)](X) = V(t',X)$$

Also, if the current spot price is X_0, then the solution to this PDE at the coordinates $t=t_0$, $X=X_0$ will yield the present value of the derivative:

$$PV = V(t_0, X_0)$$

Discretization

Since the PDE is defined continuously in the domain $[t_0,T] \times [0,\infty[$, it is necessary to discretize such domain in order to process the PDE numerically.

A square grid of points will be prepared first:

$$\{t_0, t_1, \dots, t_{n-1}, t_n\} \times \{X_1, X_2, \dots, X_{m-1}, X_m\}$$

Time axis discretization

The time points $\{t_i\}_{i=0,\dots,n}$ will be selected following these rules:

- $t0$ will match the valuation date.
- tn will match the expiry date of the option being valuated, T.
- $t0<t1\dots<tn-1<tn$.
- For each relevant date in the option (i.e., a fixing date, or a singular exercise date), such date will be part of the set $\{t_i\}_{i=0,\dots,n}$.
- The number of elements in the set will be at least the number indicated by the model parameter XX_FD_MIN_STEPS.
- The distance between two successive time steps, t_i and t_{i+1}, will not be bigger than the number of days indicated by the model parameter TV_FD_MAX_STEP_DAYS.
- The set will have as few elements as possible, provided that the above rules are respected.

Spot axis discretization

The spot points $\{X_j\}_{j=1,\dots,m}$ will be selected following these rules:

- $X1 < X2 \dots < Xm-1 < Xm$
- $X1 \leq K\alpha$
- $Xm \geq K1-\alpha$
- $X1 \leq X0 \leq Xm$, where $X0$ is the current spot
- For each relevant barrier level in the option (i.e., the level of an up continuous barrier, or a down continuous barrier), such level will be part of the set $\{X_j\}_{j=1,\dots,m}$.

$$\log X_j - \log X_{j-1} \leq \sqrt{\frac{VVT\ /\ n}{\mathrm{XX\_FD\_DV2T\_TO\_DX2\_RATIO}}}$$

- The set will have as few elements as possible, provided that the above rules are respected.
- The distance between consecutive elements in the set $\{\log X_j\}_{j=1,\dots,m}$ will be as regular as possible, provided that the above rules are respected.

where:

- $VV\overline{\overline{\overline{T}}}$ is the average variance for the period $[t_0,T]$. Such total variance is quickly approximated using equation (11.5) from 10.2 book, feed with the 25Δ Put, ATM, 25Δ $Call$ volatility points at the expiry of the option.
- K_p is the strike that verifies: $Probability(X_T \leq K_p)=p$, where the probability is calculated using the model itself. Being $P(T,K)$ the model adapted price of the put con expiry T and strike K:

$$Probability(X_T \le K_p) = e^{\int_{t_0}^{T} r(s)ds} \cdot \frac{\partial P(T,K)}{\partial K}(K_p)$$

α has been stablished to α=0.001=0.1%, which ensures that the range of the grid in the spot direction is big enough to comprise all scenarios that can have a meaningful impact in the price of the derivative.

Please note that the subindex X_0 has been avoided in the spot coordinates, to avoid confusion with the current spot. Indeed, there is no guarantee that the spot coordinates will include the current spot X_0. The most likely situation will consist of the existence of a certain index $j \in \{1,\ldots,m-1\}$ such that $X_j < X_0 < X_{j+1}$.

Discrete PDE

The algorithm will try to find a solution for the equation at the grid points. The solution will be denoted as $\{V_{ij}\}_{i=0,\ldots,n,;j=1,\ldots,m}$, being V_{ij} the estimated solution of the equation at coordinates (t_i,X_j), i.e. $V(t_i,X_j)\approx V_{ij}$.

The algorithm will proceed backwards. It will start with the solution at time t_n, which is typically known as it reduces to the payout of the derivative at $T=t_n$. From the solution at $t_n \{V_{nj}\}_{j=1,\ldots,m}$ it will compute the solution at $t_{n-1} \{V_{n-1j}\}_{j=1,\ldots,m}$. Then from the solution at t_{n-1} it will compute the solution at $t_{n-2} \{V_{n-2j}\}_{j=1,\ldots,m}$. The same process will be repeated until eventually finding the solution at time $t_0 \{V_{0j}\}_{j=1,\ldots,m}$.

The computation of solution at time t_i will be based entirely and exclusively in the solution at time t_{i+1} plus, perhaps, some extra boundary conditions. The equation to drive such calculation will be a discretized version of the original PDE:

$$-FD_{ij}^{t}(\{V_{ij}\}) \quad = \quad \frac{1}{2}\,\sigma_{ij}^2\, X_j^2\, FD_{ij}^{X^2}(\Theta,\{V_{ij}\}) \quad + \quad \tilde{\mu}_i\, X_j\, FD_{ij}^{X^1}(\Theta,\{V_{ij}\}) \quad - \; r_i\, FD_{ij}^{X^0}(\Theta,\{V_{ij}\})$$

where the FD formulas are finite difference approximations of the partial derivatives appearing in the original equation as follows:

- $FD_{ij}^{t}(\{V_{ij}\}) = \frac{V_{i+1j}-V_{ij}}{t_{i+1}-t_i}$
- $FD_{ij}^{X^0}(\Theta,\{V_{ij}\}) = \Theta\, V_{i,j} + (1-\Theta)\, V_{i+1,j}$
- $FD_{ij}^{X^1}(\Theta,\{V_{ij}\}) = \begin{cases} j=1: & \Theta\, FD_1(X_j,X_{j+1},X_{j+2},V_{i,j},V_{i,j+1},V_{i,j+2},X_j) + (1-\Theta)\, FD_1(X_j,X_{j+1},X_{j+2},V_{i+1,j},V_{i+1,j+1},V_{i+1,j+2},X_j) \\ 1<j<m: & \Theta\, FD_1(X_{j-1},X_j,X_{j+1},V_{i,j-1},V_{i,j},V_{i,j+1},X_j) + (1-\Theta)\, FD_1(X_{j-1},X_j,X_{j+1},V_{i+1,j-1},V_{i+1,j},V_{i+1,j+1},X_j) \\ j=m: & \Theta\, FD_1(X_{j-2},X_{j-1},X_j,V_{i,j-2},V_{i,j-1},V_{i,j},X_j) + (1-\Theta)\, FD_1(X_{j-2},X_{j-1},X_j,V_{i+1,j-2},V_{i+1,j-1},V_{i+1,j},X_j) \end{cases}$
- $FD_{ij}^{X^2}(\Theta,\{V_{ij}\}) = \begin{cases} j=1: & \Theta\, FD_2(X_j,X_{j+1},X_{j+2},V_{i,j},V_{i,j+1},V_{i,j+2}) + (1-\Theta)\, FD_2(X_j,X_{j+1},X_{j+2},V_{i+1,j},V_{i+1,j+1},V_{i+1,j+2}) \\ 1<j<m: & \Theta\, FD_2(X_{j-1},X_j,X_{j+1},V_{i,j-1},V_{i,j},V_{i,j+1}) + (1-\Theta)\, FD_2(X_{j-1},X_j,X_{j+1},V_{i+1,j-1},V_{i+1,j},V_{i+1,j+1}) \\ j=m: & \Theta\, FD_2(X_{j-2},X_{j-1},X_j,V_{i,j-2},V_{i,j-1},V_{i,j}) + (1-\Theta)\, FD_2(X_{j-2},X_{j-1},X_j,V_{i+1,j-2},V_{i+1,j-1},V_{i+1,j}) \end{cases}$

Θ is the model parameter XX_FD_THETA controlling the particular scheme being applied.

The core finite difference formulas FD_1 and FD_2 are:

$$FD_1(x_1, x_2, x_3, y_1, y_2, y_3, x^*) = \sum_{k=1}^{3} \frac{2x^* - \sum_{l \neq k} x_l}{\prod_{l \neq k}(x_k - x_l)} y_k$$

$$FD_2(x_1, x_2, x_3, y_1, y_2, y_3) = \sum_{k=1}^{3} \frac{2}{\prod_{l \neq k}(x_k - x_l)} y_k$$

These formulas, apart from being the most accurate any-three-point based finite difference approximations for first and second derivatives, have the following very important properties:

- $FD_1(x_1, x_2, x_3, c, c, c, x^*) = 0$
- $FD_1(x_1, x_2, x_3, cx_1, cx_2, cx_3, x^*) = c$
- $FD_1(x_1, x_2, x_3, cx_1^2, cx_2^2, cx_3^2, x^*) = 2cx^*$
- $FD_2(x_1, x_2, x_3, c, c, c) = 0$
- $FD_2(x_1, x_2, x_3, cx_1, cx_2, cx_3) = 0$
- $FD_2(x_1, x_2, x_3, cx_1^2, cx_2^2, cx_3^2) = 2c$

which means that they will produce exact approximations of first and second derivatives for any polynomial of degree 2 or below.

Additionally, it is apparent that both $FD1$ and FD_2, when applied over a pre-specified set of arguments x_1, x_2, x_3, x_*, they can be interpreted as a linear combination of the other arguments y_1, y_2, y_3.

Another consistency property is that FD_2 can be seen as the result of compounding FD_1:

Using all the indicated formulas and properties, and going back to the discretized PDE equation for a given time index i^*, and for all $j \in \{1, \dots, m\}$, there are a few realizations to make:

- There are exactly m of such equations.
- The only points participating in all those equations are exactly $\{V_{i^*,j}\}_{j=1,\dots,m} \cup \{V_{i^*+1,j}\}_{j=1,\dots,m}$, which is $2m$ points.
- If we assume known the points $\{V_{i^*+1,j}\}_{j=1,\dots,m}$, and unknown the points $\{V_{i^*,j}\}_{j=1,\dots,m}$, then we are left exactly with m equations and m unknowns.
- Inspecting it more closely, it can be confirmed that all the equations are linear. Indeed, the left hand side of such linear equations can be found as a linear combination of the known terms as well.

In other words, all the equations can be arranged together as:

$$A_i^I \cdot \overrightarrow{V_i} = A_i^E \cdot \overrightarrow{V_{i+1}}$$

Eq. 2

where:

- $\vec{V_i}$ is the vector $\begin{pmatrix} V_{i,1} \\ \vdots \\ V_{i,m} \end{pmatrix}$
- A_i^I, A_i^E are both $m \times m$ matrixes

Going a little further in the exploration of the shape of A_i^I, A_i^E, it can be observed that the following is true:

- $A_i^I = \frac{1}{\Delta t_i} I - \Theta L_i$
- $A_i^I = \frac{1}{\Delta t_i} I + (1 - \Theta) L_i$

where Δt_i is simply $\Delta t_i = t_{i+1} - t_i$, I is the identity matrix of size $m \times m$, and L_i is an $m \times m$ matrix that is the same for both sides of the equation.

Finally, just applying that simplification to Eq. 2 and multiplying both sides by Δt_i:

Finally, just applying that simplification to Eq. 2 and multiplying both sides by Δt_i:

$$(I - \Theta \, \Delta t_i \, L_i) \cdot \vec{V_i} = (I + (1 - \Theta) \, \Delta t_i \, L_i) \cdot \overrightarrow{V_{i+1}}$$

Eq. 3

indeed:

$$\vec{V_i} = (I - \Theta \, \Delta t_i \, L_i)^{-1} \cdot (I + (1 - \Theta) \, \Delta t_i \, L_i) \cdot \overrightarrow{V_{i+1}}$$

and defining the $m \times m$ matrix $\ddot{E}_{t_i \leftarrow t_{i+1}}$:

$$\ddot{E}_{t_i \leftarrow t_{i+1}} = (I + \Theta \, \Delta t_i \, L_i)^{-1} \cdot (I + (1 - \Theta) \, \Delta t_i \, L_i)$$

the equation takes the expectation evocating form:

$$\vec{V_i} = \ddot{E}_{t_i \leftarrow t_{i+1}} \cdot \overrightarrow{V_{i+1}}$$

The most important comment to make in face of the shape in Eq. 3, is that the parameter Θ has a deep impact on it. For instance:

- When $\Theta = 0$, the equation becomes $\vec{V_i} = (I + (1 - \Theta) \, \Delta t_i \, L_i) \cdot \overrightarrow{V_{i+1}}$. This is known as the *fully explicit* scheme, since there is no linear system solution involved in the calculation of $\vec{V_i}$.
- When $\Theta = 1$, the equation becomes $(I - \Theta \, \Delta t_i \, L_i) \cdot \vec{V_i} = \overrightarrow{V_{i+1}}$. This is known as the *fully implicit* scheme, since the solution of a linear system of equations is the only step required for the calculation of $\vec{V_i}$.
- When $0 < \Theta < 1$, the scheme has both, an explicit and an implicit step. In particular, when $\Theta = 0.5$, the scheme is the known Crank-Nicolson.

A good study of the properties of each scheme can be found in the works indicated in the section 9.2. As a rule of thumb, $\Theta=1$ is the most stable configuration, $\Theta=0.5$ is the most accurate, and $\Theta=0$ is the fastest. It should be stressed that values of $0 \leq \Theta < 0.5$ are very likely to produce big instabilities in the solution, and should not be used in general.

Another important observation to make is that the matrix L_i can be decomposed as:

$$L_i = \frac{1}{2}\Sigma_i^2 \cdot X^2 \cdot L^{X^2} + \mu_i \cdot X \cdot L^{X^1} - r_i \cdot I$$

where:

- X is the $m \times m$ diagonal matrix: $X = \begin{pmatrix} X_1 & 0 & \dots & 0 \\ 0 & X_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & X_m \end{pmatrix}$
- Σ_i is the $m \times m$ diagonal matrix: $\Sigma_i = \begin{pmatrix} \sigma_{i1} & 0 & \dots & 0 \\ 0 & \sigma_{i2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \sigma_{im} \end{pmatrix}$
- L^{X^1} is the $m \times m$ matrix that verifies: $L^{X^1} \cdot \vec{V} = \begin{pmatrix} FD_1(X_1, X_2, X_3, V_1, V_2, V_3, X_1) \\ FD_1(X_1, X_2, X_3, V_1, V_2, V_3, X_2) \\ \vdots \\ FD_1(X_{j-1}, X_j, X_{j+1}, V_{j-1}, V_j, V_{j+1}, X_j) \\ \vdots \\ FD_1(X_{m-2}, X_{m-1}, X_m, V_{m-2}, V_{m-1}, V_m, X_{m-1}) \\ FD_1(X_{m-2}, X_{m-1}, X_m, V_{m-2}, V_{m-1}, V_m, X_m) \end{pmatrix}$
- L^{X^2} is the $m \times m$ n matrix that verifies: $L^{X^2} \cdot \vec{V} = \begin{pmatrix} FD_2(X_1, X_2, X_3, V_1, V_2, V_3) \\ FD_2(X_1, X_2, X_3, V_1, V_2, V_3) \\ \vdots \\ FD_2(X_{j-1}, X_j, X_{j+1}, V_{j-1}, V_j, V_{j+1}) \\ \vdots \\ FD_2(X_{m-2}, X_{m-1}, X_m, V_{m-2}, V_{m-1}, V_m) \\ FD_2(X_{m-2}, X_{m-1}, X_m, V_{m-2}, V_{m-1}, V_m) \end{pmatrix}$

Using that expression, and the properties of the FD_1 and FD_2 formulas, it is clear that:

$$L_i \cdot \vec{U} = -r_i \cdot \vec{U}$$

$$L_i \cdot \vec{X} = (\tilde{\mu}_i - r_i) \cdot \vec{X}$$

where:

- $\vec{U}$ is the $m \times 1$ vector $\vec{U} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$
- $\vec{X}$ is the $m \times 1$ vector $\vec{X} = \begin{pmatrix} X_1 \\ \vdots \\ X_m \end{pmatrix}$

Using that, and imposing the desirable properties that the solver step $t_{i+1} - t_i$ yields a perfect solution for the expectation of the basic payout 1 at t_{i+1}. Defining DF_i as the solution of the continuous PDE at t_i of the basic payout 1 at t_{i+1}:

$$E_{t_i \leftarrow t_{i+1}}[1](X) = DF_i = \exp\left(-\int_{t_i}^{t_{i+1}} r(s)ds\right)$$

The discretized version of the PDE will verify it if and only if:

$$\dddot{E}_{t_i \leftarrow t_{i+1}} \cdot \vec{U} = DF_i \cdot \vec{U}$$
$$\Updownarrow$$
$$DF_i \cdot \vec{U} = \dddot{E}_{t_{i+1} \leftarrow t_i} \cdot \vec{U}$$
$$\Updownarrow$$
$$DF_i \cdot \vec{U} = (I - \Theta\, \Delta t_i\, L_i)^{-1} \cdot (I + (1-\Theta)\, \Delta t_i\, L_i) \cdot \vec{U}$$
$$\Updownarrow$$
$$(I - \Theta\, \Delta t_i\, L_i) \cdot DF_i \cdot \vec{U} = (I + (1-\Theta)\, \Delta t_i\, L_i) \cdot \vec{U}$$
$$\Updownarrow$$
$$(1 + \Theta\, \Delta t_i\, r_i) \cdot DF_i \cdot \vec{U} = (1 - (1-\Theta)\, \Delta t_i\, r_i) \cdot \vec{U}$$
$$\Updownarrow$$
$$(1 + \Theta\, \Delta t_i\, r_i) \cdot DF_i = (1 - (1-\Theta)\, \Delta t_i\, r_i)$$
$$\Updownarrow$$
$$r_i = \frac{1}{\Delta t_i} \frac{1 - DF_i}{1 - \Theta + \Theta \cdot DF_i}$$

Please observe that, as long as $0 \leq \Theta \leq 1$, and $0 < DF_i$, r_i will be well defined, and will also preserve the sign of the average discount rate implied by DF_i:

$$\begin{cases} DF_i < 1 \Rightarrow r_i > 0 \\ DF_i = 1 \Rightarrow r_i = 0 \\ DF_i > 1 \Rightarrow r_i < 0 \end{cases}$$

A similar derivation can be done for the equally basic payout $X_{t_{i+1}}$ at t_{i+1}. The solution of the continuous PDE is:

$$E_{t_i \leftarrow t_{i+1}}[X](X) = X \cdot \exp\left(-\int_{t_i}^{t_{i+1}} (r(s) - \tilde{\mu}(s))ds\right)$$

Defining $DF_i^Q = \exp(-\int_{t_i}^{t_{i+1}} (r(s) - \tilde{\mu}(s))ds)$ and imposing again that the solution works in the discretized version of the PDE:

$$\dddot{E}_{t_i \leftarrow t_{i+1}} \cdot \vec{X} = DF_i^Q \cdot \vec{X}$$
$$\Updownarrow$$
$$\vdots$$
$$\Updownarrow$$
$$\tilde{\mu}_i = r_i - q_i, \qquad q_i = \frac{1}{\Delta t_i} \frac{1 - DF_i^Q}{1 - \Theta + \Theta \cdot DF_i^Q}$$

Therefore, in the discretized version of the PDE, r_i and $\tilde{\mu}_i$ can be chosen so that the methodology matches perfectly the solution of the trivial payouts 1 and X under the continuous PDE.

Regarding σ_{ij}^2, arguments for calibrating it can be summoned as well. For instance, in the constant volatility case (i.e., Theoretical model), the $E_{t_i \leftarrow t_{i+1}}[X^2](X)$ can be used to discover a unique σ_{ij}^2 for each step $t_i \rightarrow t_{i+1}$. In the local volatility case, the σ_{ij}^2 could be calibrated to the known prices of vanilla options. However, neither of those relationships are used in the final implementation. Contrary to what happens with r_i and q_i, the proper sign of σ_{ij}^2 cannot be guaranteed, nor the stability of the solution. Therefore, these elements are initialized directly from their continuous counterparts:

$$\sigma_{ij}^2 = \frac{1}{\Delta t_i} \int_{t_i}^{t_{i+1}} \sigma_{LV}(s, X_j)^2 ds$$

Numerical routine

The only thing that is left in order to be able to compute the solution of the discretized PDE, is to execute each of the steps:

$$\vec{V_i} = \ddot{E}_{t_i \leftarrow t_{i+1}} \cdot \overrightarrow{V_{i+1}}$$
$$\Updownarrow$$
$$(I - \Theta\, \Delta t_i\, L_i) \cdot \vec{V_i} = (I + (1 - \Theta)\, \Delta t_i\, L_i) \cdot \overrightarrow{V_{i+1}}$$

Since $m \times m$ matrixes are involved, in principle this is a task with a computational cost of $O(m_3)$.

However, an even closer look at the matrix L_i, will reveal that it is almost tridiagonal (excepting in the first and last row, where one extra space next to the third diagonal is occupied). This means that:

- The computation $\overrightarrow{V_{i+\frac{1}{2}}} := (I + (1 - \Theta)\, \Delta t_i\, L_i) \cdot \overrightarrow{V_{i+1}}$ is $O(m)$.
- The computation $\vec{V_i} = (I - \Theta\, \Delta t_i\, L_i)^{-1} \cdot \overrightarrow{V_{i+\frac{1}{2}}}$ is also $O(m)$, since that's the algorithmic complexity of solving the implied linear system of equations using an LU algorithm smart about the location of zeros in the matrix $(I - \Theta\, \Delta t_i\, L_i)$.

So the step to calculate $\vec{V_i}$ from $\overrightarrow{V_{i+1}}$ will have an overall algorithmic complexity of $O(m)$.

Combining the n successive steps to compute the solution at time t_0 will naturally have an overall complexity of $O(n \times m)$.

Remarks

The outlined algorithm is quite standard, excepting for a couple particular ways around some topics.

Uniformity of space grid

Most textbooks work with uniform space grids. The space grid presented here is mainly obeying the relationship:

$$\log X_j - \log X_{j-1} = c, \forall\, j \in \{1, \dots, m\}$$

Or in other words, it is not a grid uniform in X, but in x, where $x = \log X$.

If using a PDE working on $t \times x$ coordinates, as opposed to $t \times X$ coordinates, it is easy to see that the PDE is transformed into:

$$-\frac{\partial W(t,x)}{\partial t} = \frac{1}{2}\, \sigma_{LV}(t, e^x)^2 \frac{\partial^2 W(t,x)}{\partial x^2} + \left(\tilde{\mu}(t) - \frac{1}{2}\, \sigma_{LV}(t, e^x)^2\right) \frac{\partial W(t,x)}{\partial x} - r(t)\, W(t,x)$$

where $W(t,x) = V(t, e_x)$.

This would be the PDE to apply in the $t \times x$ space, where the uniformity that most theories work with is present.

However, just applying the identities:

$$\begin{aligned}
W(t,x) &= V(t,X) \\
\frac{\partial W(t,x)}{\partial t} &= \frac{\partial V(t,X)}{\partial t} \\
\frac{\partial W(t,x)}{\partial x} &= X \cdot \frac{\partial V(t,X)}{\partial X} \\
\frac{\partial^2 W(t,x)}{\partial x^2} &= X^2 \cdot \frac{\partial^2 V(t,x)}{\partial x^2} + X \cdot \frac{\partial V(t,X)}{\partial X}
\end{aligned}$$

the new equation is no other that Eq. 1, exactly.

So in a sense, what is being done can be interpreted as having a mostly uniform grid in $x=\log X$, having the appropriate PDE in $t \times x$ coordinates, then replacing the partial derivatives in the PDE by its equivalent in the $t \times X$ space.

Matching trivial solutions

Having replaced the derivatives in x by its equivalent derivatives in X, coupled with the used maximum order three-point finite difference formulas to approximate $\frac{\partial V(t,X)}{\partial X}$ and $\frac{\partial^2 V(t,x)}{\partial x^2}$, that matching the solutions of the trivial payouts 1 and X reduces to a very simple exercise, yielding the appropriate r_i and $\tilde{\mu}_i$ coefficients to use in the discretized version of the PDE.

And it is not only that r_i and $\tilde{\mu}_i = r_i - q_i$ are discovered, but also that r_i and q_i take natural values if compared with their continuous version counterparts.

Boundary conditions

No explicit mention to boundary conditions have been done yet. And that's because they have been naturally integrated in the general algorithm.

In one hand, special approximations for first and second order derivatives have been used at the end points of the space grid.

On the other hand, the numerical algorithm is just slightly more complex than usual Thomas routine for the solution of tridiagonal systems of equations, and still based in the same principles (i.e., LU decomposition), in order to accommodate the two extra terms that appear in the linear matrixes defining the solution.

The resulting management of boundary conditions, is equivalent to the BC2 type boundary conditions introduced in the *Boundary Conditions* section in chapter 4 of *Tavella & Randall*.

It is also equivalent to what is explained in sections 2.2.2 and 10.1.5.2 of *Andersen & Piterbarg*.

Needless to say, in the presence of barriers, Dirichlet boundary conditions are applied instead.

Scheme order

The order of convergence of the scheme in the t direction will be as usual for Generalized Euler scheme:

$$\begin{cases} O(\Delta t) \; if \; \Theta \neq 0.5 \\ O((\Delta t)^2) \; if \; \Theta = 0.5 \end{cases}$$

The order of convergence of the scheme in the X direction will be $O(\Delta \log X)$, which can seem rather low order, but in practice is not behaving much worse than a pure $O((\Delta \log X)^2)$ scheme.

Section 11. PricerFXOHestonFiniteDifference

PricerFXOHestonFiniteDifference calculates prices for FX Options under the Heston model via a Finite Difference integration.

The modelling assumptions are exactly the same as those exposed in Section 6 for PricerFXOHeston, and won't be indicated here. Therefore, both pricers should produce very similar results, except that the numerical methodology used for the calculation is different.

This pricer is not expected to be used in practice, since a more sophisticated alternative exists in FXOLocalStochasticVolatilityFiniteDifference, and a much faster one exists for simple payouts under FXOHeston.

However, it can be effectively used for validation purposes: the Finite Difference algorithm is the basis of some more sophisticated models available, and many of the valuation steps are shared among such sophisticated applications of Finite Difference and this one. The matching of this pricer against existing Heston analytical formulas should generate confidence on the general application of Finite Differences in FXO modeling.

11.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
H_REFERENCE_DELTA	**Double** Delta of the wing vanillas to be used for calibration. Expressed in % directly. A value strictly between 0 and 50 is expected.	25
H_MEAN_REVERSION_1Y	**Double** The mean reversion speed to use foroptions expiring in one year time. For options with expiry date differentto one year, the mean reversion is calculated as: $\kappa_{TY} = \kappa_{1Y} / \mathrm{T}$ where T is the time to expiry measured in ACT/365 years.	1
H_FD_MAX_STEP_DAYS	**Double** The maximum size in days allowed for one Finite Difference time step. A smaller maximum size will increase the precision and accuracy of the results.	7
H_FD_MIN_STEPS	**Integer**	75

Pricing Parameter	Description	Typical Value
	The minimum number of time steps to apply, regardless of the expiry of the option. A bigger number of minimum steps will increase the precision and accuracy for short term options.	
H_FD_DV2T_TO_DX2_RATIO	**Double** The average ratio of the size of the time step to the square of the size of the space step. More precisely: $$ratio = \frac{\sigma^2 \Delta t}{(\Delta \log S)^2}$$ This ratio is used to decide the size of the space step: $$\Delta \log S = \sqrt{\frac{\sigma^2 \Delta t}{ratio}}$$ A lower ratio will reduce the computational cost, and the precision in the space direction. A too high ratio will make oscillations in the solution more likely to appear.	0.5
H_FD_THETA	**Double** The theta defining the Generalized Euler scheme to apply. Theta must be a number between 0 and 1. The following numbers will produce the named schemes: - 0: Explicit Euler - 0.5: Crank-Nicolson - 1: Implicit Euler	0.5
H_FD_SPLIT	**Boolean** Whether a splitting methodology should be used when solving the 2- Dimensional Finite Difference problem.	true

Pricing Parameter	Description	Typical Value
	A splitting methodology will make the calculation faster, at the expense of some accuracy.	
H_FD_SPLIT_PRED_CORR_ITERS	**Integer** It only applies when H_FD_SPLIT is set to true. The number of times that the split method will iterate to estimate the unknown implied component. A value of at least 1 is recommended, unless the model correlation is 0 as well.	1
H_FD_SPLIT_IMITATE_3RD_ORDER	**Boolean** It only applies when H_FD_SPLIT is set to true, and H_FD_SPLIT_PRED_CORR_ITERS is greater than 0. If set to false, the unknown third order components will be discarded when iterating to compute the solution. If set to true, the estimated third order components will be included in the predictor-corrector iteration. Setting it to true, while not enhancing the quality of the results, will make them closer to what it would be obtained shall H_FD_SPLIT set to *false*.	True
H_FD_MAX_THREADS	**Integer** The maximum number of threads that can be used by the model. The larger the number, the faster the pricing will be cleared. A number higher than the computer physical processor count will typically produce no marginal improvement. Parallelization occurs only at a per- measure level, so there will be no appreciable speed gain when applied to the calculation of PV only.	1
H_FD_TV_MODEL	**Domain** The model to be used for computation of TV. Two main methods are supported:	Theoretical

Pricing Parameter	Description	Typical Value
	- Theoretical: use same model as FXOTheoretical pricer. - TheoreticalFiniteDifference: use same model as FXOTheoreticalFiniteDifference pricer.	

11.1.1 Configuration of Craig-Sneyd scheme

The classical split scheme from Craig-Sneyd can be configured by setting:

- H_FD_SPLIT = true
- H_FD_SPLIT_PRED_CORR_ITERS = 1
- H_FD_SPLIT_IMITATE_3RD_ORDER = false

The Craig-Sneyd scheme as presented in the original paper is defined using two different parameters: Θ and λ.The indicated configuration will produce the Craig-Sneyd scheme with $\Theta = \lambda = H\_FD\_THETA$. The case when $\Theta \neq \lambda$ is not supported.

11.1.2 Logging key

If the logging key *Analytics.FXO.Heston* is active, a trace with details about the calibration results will appearin the log.

11.2 References

Tavella, D. & Randall, C. Pricing Financial Instruments: The Finite Difference method. John Wiley & Sons, Inc.(2000)

Clark, Iain J. Foreign exchange option pricing : a practitioner's guide. John Wiley & Sons Ltd (2011)

Austing, P. Smile Pricing Explained. Palgrave Macmillan (2014)

Craig, I. & A. Sneyd. An alternating direction implicit scheme for parabolic equations with mixed derivatives.Comput. Math. Applic (1988)

11.3 Implementation Details

As indicated in Section 12, where the Local Stochastic Volatility model is introduced, the Heston model can be seenas a particular case of the Local Stochastic Volatility model.

Therefore, any technique valid to compute the price under Local Stochastic Volatility model, can be easily reused tothe same effect under Heston model.

Indeed, the implementation of Finite Difference solving is shared among both pricers: HestonFiniteDifference andLocalStochasticVolatilityFiniteDifference.

The reader is kindly invited to refer to the Implementation Details detailed in Section 12.3, where explanation isprovided about the Finite Difference algorithm used for Local Stochastic Volatility (and Heston) model.

Section 12. PricerFXOLocalStochasticVolatilityFiniteDifference

PricerFXOLocalStochasticVolatilityFiniteDifference implements the solution of Local Stochastic Volatility model viaFinite Differences.

The **L**ocal **S**tochastic **V**olatility model (LSV going forward) is a combination of the principles behind the **L**ocal **V**olatility model (LV going forward), and usual **S**tochastic **V**olatility models (SV going forward), like Heston. Theaim is to get the advantages of both approaches, and not suffer so much from its disadvantages.

The Local Volatility model has the theoretical capability of matching the prices of all vanilla options, but in practiceit produces smile dynamics that are not realistic with what is observed in FX markets.

A Stochastic Volatility model is richer at the time of representing the market dynamics, however its reduced number of parameters make difficult to match the prices of vanillas across all the range. In a sense, it imposes ahard choice when calibrating its parameters: should they be calibrated as to match the vanilla prices as much aspossible, or should they calibrated as to represent the true market dynamics?

The LSV model tries to conciliate both aims. Its **S**tochastic **D**ifferential **E**quation (SDE going forward), in the mostgeneral form that will be managed here is:

$$\frac{dX_t}{X_t} = \mu(t)dt + A_{LV}(t, X_t) \cdot B_{SV}(t, v_t)dW_t^X$$

$$dv = C(t, v_t)dt + D(t, v_t)\, dW_t^v$$

$$\langle dW_t^X, dW_t^v \rangle = \rho(t)dt$$

where:

- Xt is the observed exchange rate at time t.
- v_t is the instantaneous variance at time t.
- $\mu(t)$ is a curve of known-in-advance, instantaneous, drifts for the returns of X_t.
- $A_{LV}(t,X_t)$, $B_{SV}(t,v_t)$, $C(t,v_t)$, $D(t,v_t)$ are smooth functions of their parameters.
- $\rho(t)$ is the instantaneous correlation between the two driving Brownian motions W_{tX} and W_{tv}.

In other words, the natural diffusion coefficient for the spot equation, can be decomposed in two parts:

$$\sigma(t, X_t, v_t) = A_{LV}(t, X_t) \cdot B_{SV}(t, v_t)$$

where $A_{LV}(t,X_t)$ is regarded as the *Local Volatility* component, and $B_{SV}(t,v_t)$ is regarded as the *Stochastic Volatility* component. This factorization, while not entirely flexible in the shapes that $\sigma(t,X_t,v_t)$ can take, is just flexible enough to attain the desired goals:

- It is possible to calibrate A_{LV} in such a way that the vanilla prices are matched perfectly.
- B_{SV} will follow the dynamics of v_t, an additional factor that ideally will be modelled after the characteristic of the market being considered.

The particular implementation in Calypso will use the following definitions for $B_{SV}(t,v_t)$, $C(t,v_t)$, $D(t,v_t)$,$\rho(t)$:

- $B_{SV}(t,v_t) = \sqrt{v_t}$
- $C(t,v_t) = \kappa \cdot (\theta - v_t)$
- $D(t,v_t) = \lambda_\eta \cdot \eta \cdot \sqrt{v_t}$
- $\rho(t) = \lambda_\rho \cdot \rho$

where:

- θ,κ,η,ρ are calibrated following the same strategy as in the pricer FXOHeston (see section 6.3).
- $\lambda\eta$ is the stochastic volatility weight, in the range [0,1].
- λ_ρ is the local volatility weight, in the range $[0,1]$ as well.

That is, the base dynamics chosen for the stochastic vol factor, are Heston dynamics. While many other dynamics would have been possible, and perhaps easier to implement, Heston has the advantage of being a popular, consolidated, and somewhat standard model.

The extra parameters λ_η, λ_ρ allow to manually tune in or out important characteristics of Heston, like the vol of vol, or the correlation between spot and vol.

Indeed, it can be easily checked that by choosing $\lambda_\eta=1$, $\lambda_\rho=1$ and $A_{LV}(t,X_t)=1$, exactly the same model that was introduced in Section 6 and reused in Section 11 is implied.

However, at this point, while stochastic volatility dynamics are already part of the model, and the user has extra agency on their impact, the perfect calibration to vanillas remains to be achieved. Filling this last gap is the exact role reserved for $ALV(t,Xt)$.

One first insight into this problem can be achieved by setting the parameter $\lambda_\eta=0$. In this case:

$$dv = C(t, v_t)dt$$

And therefore v_t can be discovered by solving the differential equation $\frac{dv(t)}{dt} = C\big(t, v(t)\big)$, $v(0) = v_0$. The equation for X becomes:

$$\frac{dX_t}{X_t} = \mu(t)dt + A_{LV}(t, X_t) \cdot B_{SV}\big(t, v(t)\big)dW_t^X$$

and clearly, it is enough with imposing $A_{LV}(t, X_t) \cdot B_{SV}\big(t, v(t)\big) = \sigma_{LV}(t, X_t)$ to calibrate $A_{LV}(t, X_t)$. Simply:

$$A_{LV}(t, X_t) = \frac{\sigma_{LV}(t, X_t)}{B_{SV}\big(t, v(t)\big)}$$

The case with $\lambda_\eta \neq 0$ is clearly not so easy. But a good understanding of Dupire's theory about Local Volatility, distils this as the one and only (theoretical) constraint in order to match the prices of vanillas:

$$\sigma_{LV}{}^2(t, K) = E[\sigma^2(t, X_t, v_t)|X_t = K]$$

where $\sigma_{LV}(t,K)$ is the Local Volatility's volatility at time t and spot K (see Section 8).

Using $\sigma(t,X_t,v_t)=A_{LV}(t,X_t)\cdot B_{SV}(t,v_t)$:

$$\sigma_{LV}{}^2(t,K) = E[\sigma^2(t,X_t,v_t)|X_t = K]$$
$$\Updownarrow$$
$$\sigma_{LV}{}^2(t,K) = E[A_{LV}^2(t,X_t) \cdot B_{SV}^2(t,v_t)|X_t = K]$$
$$\Updownarrow$$
$$\sigma_{LV}{}^2(t,K) = A_{LV}^2(t,K) \cdot E[B_{SV}^2(t,v_t)|X_t = K]$$
$$\Updownarrow$$
$$A_{LV}(t,K) = \sqrt{\frac{\sigma_{LV}{}^2(t,K)}{E[B_{SV}^2(t,v_t)|X_t = K]}}$$

Therefore, provided that all other details of the model are known, the Local Volatility theory gives a direct relationship between the Local Volatility $\sigma_{LV}(t,K)$ and $A_{LV}(t,K)$. Calibrating A_{LV} to fulfill this relationship at every time t and strike K, will make the LSV match the vanilla prices.

The LSV model is clearly more sophisticated than LV and SV, but its unique ability to overcome the most obvious limitations of each of those models, and the fact that FX markets present particularly regular volatilities (which translates in more stable calibration), have made it the standard choice for valuation of FX exotics.

12.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
H_REFERENCE_DELTA	**Double** Delta of the wing vanillas to be used for calibration. Expressed in % directly. A value strictly between 0 and 50 is expected.	25
H_MEAN_REVERSION_1Y	**Double** The mean reversion speed to use foroptions expiring in one year time. For options with expiry date differentto one year, the mean reversion is calculated as: $\kappa_{TY} = \kappa_{1Y} / \mathrm{T}$ where T is the time to expiry measuredin ACT/365 years.	1
LSV_STOCHASTIC_VOL_WEIGHT	**Double** The mixing parameter, that tunes the behaviour of the model. It should be given a value between 0.0 and 1.0.	0.60

Pricing Parameter	Description	Typical Value
	If set to 0.0, the model will behave the same way as LocalVolatility model. If set to 1.0, the model will behave in a mode quite similar to Heston. If set to an intermediate number, the model will show a behaviour that is half way between pure Local Volatility, and Heston.	
LSV_CORRELATION_WEIGHT	**Double** A parameter deciding how much of the calibrated Heston correlation should be included in the Local Stochastic Volatility model. It should be given a value between 0.0 and 1.0. If set to 0.0, the correlation will be assumed to be 0, regardless of the calibrated Heston correlation. If set to 1.0, the correlation will be assumed to be exactly the same as the one calibrated Heston correlation. For intermediate values, the respective fraction of the calibrated Heston correlation will be used. A value of 0.0 will make this model ignore the cross derivative term, which seems to be common practice in the FX option markets.	0.0
LSV_FD_MAX_STEP_DAYS	**Double** The maximum size in days allowed for one Finite Difference time step. A smaller maximum size will increase the precision and accuracy of the results.	7
LSV_FD_MIN_STEPS	**Integer** The minimum number of time steps to apply, regardless of the expiry of the option. A bigger number of minimum steps will increase the precision and accuracy for short term options.	75
LSV_FD_DV2T_TO_DX2_RATIO	**Double**	0.5

Pricing Parameter	Description	Typical Value
	The average ratio of the size of the time step to the square of the size of the space step. More precisely: $$ratio = \frac{\sigma^2 \Delta t}{(\Delta \log S)^2}$$ This ratio is used to decide the size of the space step: $$\Delta \log S = \sqrt{\frac{\sigma^2 \Delta t}{ratio}}$$ A lower ratio will reduce the computational cost, and the precision in the space direction. A too high ratio will make oscillations in the solution more likely to appear.	
LSV_FD_THETA	**Double** The theta defining the Generalized Euler scheme to apply. Theta must be a number between 0 and 1. The following numbers will produce the named schemes: - 0: Explicit Euler - 0.5: Crank-Nicolson - 1: Implicit Euler	0.5
LSV_FD_SPLIT	**Boolean** Whether a splitting methodology should be used when solving the 2- Dimensional Finite Difference problem. A splitting methodology will make the calculation faster, at the expense of some accuracy.	true
LSV_FD_SPLIT_PRED_CORR_ITERS	**Integer** It only applies when LSV_FD_SPLIT is set to true. The number of times that the split method will iterate to estimate the unknown implied component.	1

Pricing Parameter	Description	Typical Value
	A value of at least 1 is recommended, unless LSV_CORRELATION_WEIGHT is 0.	
LSV_FD_SPLIT_IMITATE_3RD_ORDER	**Boolean** It only applies when LSV_FD_SPLIT is set to true, and LSV_FD_SPLIT_PRED_CORR_ITERS is greater than 0. If set to false, the unknown third order components will be discarded when iterating to compute the solution. If set to true, the estimated third order components will be included in the predictor-corrector iteration. Setting it to true, while not enhancing the quality of the results, will make them closer to what it would be obtained shall LSV_FD_SPLIT set to false.	true
LSV_FD_MAX_THREADS	**Integer** The maximum number of threads that can be used by the model. The larger the number, the faster the pricing will be cleared. A number higher than the computer physical processor count will typically produce no marginal improvement. Parallelization occurs only at a per- measure level, so there will be no appreciable speed gain when applied to the calculation of PV only.	1
LSV_FD_TV_MODEL	**Domain** The model to be used for computation of TV. Two main methods are supported: - Theoretical: use same model as FXOTheoretical pricer. - TheoreticalFiniteDifference: use same model as FXOTheoreticalFiniteDifference pricer.	Theoretical
LSV_EXPIRY_BARRIER_MODEL	**Domain** It applies to options with EXPIRY barriers only.	Market

Pricing Parameter	Description	Typical Value
	Even the usage of LSV for FULL and PARTIAL barriers is natural, EXPIRY barriers are best calculated using the PricerFXOMarket. This parameter allows choosing a different model than LSV to be applied for EXPIRY barriers. Possible values are: - Market / Mkt: valuate EXPIRY barriers using PricerFXOMarket - LocalStochasticVolatilityFinite Difference / LSV: valuate EXPIRY barriers using LSV methodology	
LSV_THIRD_CCY_QUANTO_MODEL	**Domain** Pricer FXOLocalStochasticVolatilityFiniteDifference has not been extended to handle Quanto options. Since Calypso pricer configuration scheme doesn't allow to configure quanto options separately from other more LSV friendly types, this escape parameter has been prepared. The quanto options will be valued using the model hinted by this parameter. Possible values are (*): - Theoretical / TV: valuate quanto options using PricerFXOTheoretical. - Best: valuate quanto options using the best model available for it (which is PricerFXOTheoretical nowadays). (*) Even though there is only one possible value, filling this pricer parameter is mandatory in the case that quanto options are going to be valuated. The intention is to ensure that the user is aware of the limitations of the FXOLocalStochasticVolatilityFiniteDifference model before escaping to alternative TV valuation. (**) EXPIRY_BARRIER_MODEL will be checked first. In the event of having an assignment different from LSV model, the assigned EXPIRY_BARRIER_MODEL will be used to the evaluation of quanto expiry barriers, regardless	Market

Pricing Parameter	Description	Typical Value
	of the value carried by VV_THIRD_CCY_QUANTO_MODEL.	
LSV_ACCRUAL_RESETTABLE_MODEL	Domain Resettable Accrual options, can't be valuated under the current the FXOLocalStochasticVolatilityFiniteDifference pricer. Since Calypso pricer configuration scheme doesn't allow to configure Resettable Accrual options separately from other more Finite Difference friendly types, this escape parameter has been prepared. The Resettable Accrual options will be valued using the model hinted by this parameter. Possible values are (*): - Theoretical / TV: valuate Resettable Accrual options using PricerFXOTheoretical. - Market / Mkt: valuate Resettable Accrual options using PricerFXOMarket. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate Resettable Accrual options using PricerFXOLocalVolatilityMonteCarlo. - Best: valuate Resettable Accrual using the best model available for it (currently pointing to PricerFXOMarket). (*) Filling this pricer parameter is mandatory in the case that Resettable Accrual options are going to be valuated. The intention is to ensure that the user is aware of the limitations of the FXOLocalStochasticVolatilityFiniteDifference model before escaping to alternative valuation.	Market

12.1.1 Configuration of Craig-Sneyd scheme

The classical split scheme from Craig-Sneyd can be configured by setting:

- LSV_FD_SPLIT = true
- LSV_FD_SPLIT_PRED_CORR_ITERS = 1
- LSV_FD_SPLIT_IMITATE_3RD_ORDER = false

The Craig-Sneyd scheme as presented in the original paper is defined using two different parameters: Θ and λ.The indicated configuration will produce the Craig-Sneyd scheme with $\Theta = \lambda = H\_FD\_THETA$. The case when $\Theta \neq \lambda$ is not supported.

12.1.2Logging Key

If the logging key *Analytics.FXO.Heston* is active, a trace with details about the calibration results will appearin the log.

12.2 References

Tavella, D. & Randall, C. Pricing Financial Instruments: The Finite Difference method. John Wiley & Sons, Inc. (2000)

Clark, Iain J. Foreign exchange option pricing : a practitioner's guide. John Wiley & Sons Ltd (2011)

Austing, P. Smile Pricing Explained. Palgrave Macmillan (2014)

Craig, I. & A. Sneyd. An alternating direction implicit scheme for parabolic equations with mixed derivatives. Comput. Math. Applic (1988)

Andreasen, J., Huge, B. Random grids. Risk magazine (23 Jun 2011)

Gatheral, J. The volatility surface: a practitioner's guide, John Wiley & Sons (2006)

12.3 Implementation Details

The spirit of the implementation of the Finite Difference valuation, is the same as in Section 10.3. The main difference is the space dimensionality of the Stochastic Differential Equation, which goes from 1 (one process forspot) to two (one process for spot, and another for stochastic volatility).

The practice for implementation of 2D Finite Difference valuation, while not as documented as in the 1D case, it isstill widely documented. The books referenced in section 12.2 will help getting a fair idea of the state of the art implementations for LSV finite differences.

In the following paragraphs, detail is provided in particular decisions taken while implementing the standard technique. They are presented as a sequence of steps, relationships and properties. However, no especial effort isdone on justifying the particular decisions adopted, understanding that a quiet read of basic textbooks in the subject will be much more effective to develop some intuition about the technique.

Exploiting the fact that the Heston model can be seen as a particular case of the Local Stochastic Volatility model, the implementation details explained here will be valid for both models at the same time: Local Stochastic VolatilityFinite Difference, and Heston Finite Difference.

Partial Differential Equation

The PDE used in practice, is the Local Stochastic Volatility PDE without any transformation:

$$
\begin{aligned}
-\frac{\partial V(t,X,v)}{\partial t} = {} & \frac{1}{2} A_{LV}(t,X)^2 B_{SV}(t,v)^2 X^2 \frac{\partial^2 V(t,X,v)}{\partial X^2} + \rho(t) A_{LV}(t,X) B_{SV}(t,v) D(t,v) X \frac{\partial^2 V(t,X,v)}{\partial X\, \partial v} + \frac{1}{2} D(t,v)^2 \frac{\partial^2 V(t,X,v)}{\partial v^2} \\
& + \tilde{\mu}(t)\, X \frac{\partial V(t,X)}{\partial X} + C(t,v) \frac{\partial V(t,X,v)}{\partial v} - r(t)\, V(t,X,v)
\end{aligned}
$$

where:

- $V(t,X,v)$ is the value of the derivative at time t if the spot at time t, X_t, took the value X, and the stochastic volatility at time t, v_t, took the value v.
- $A_{LV}(t,X) = \begin{cases} 1 \text{ under Heston model} \\ \text{Result of calibration under LSV model} \end{cases}$
- $B_{SV}(t,v) = v$
- $C(t,v) = \kappa \cdot (\theta - v)$
- $D(t,v) = \lambda_\eta \cdot \eta \cdot \sqrt{v}$, with $\lambda_\eta = \begin{cases} 1 \text{ under Heston model} \\ \text{LSV\_STOCHASTIC\_VOL\_WEIGHT under LSV model} \end{cases}$
- $\rho(t) = \lambda_\rho \cdot \rho$, with $\lambda_\rho = \begin{cases} 1 \text{ under Heston model} \\ \text{LSV\_CORRELATION\_WEIGHT under LSV model} \end{cases}$

A notation to represent special solutions of this PDE will be introduced. If $PO(X,v)$, with $X \in [0,\infty[$ is used as an initial condition at $t"$ (i.e., imposing $V(t",X,v) = PO(X,v), \forall X \in [0,\infty[, \forall v$), then the particular solution of such PDE at t', with $t' < t"$, will be denoted as:

$$E_{t' \leftarrow t"}[PO(X,v)](X,v) = V(t',X,v)$$

Also, if the current spot price is X_0, and the current stochastic volatility is v_0, then the solution to this PDE at the coordinates $t = t_0$, $X = X_0$, $v = v_0$ will yield the present value of the derivative:

$$PV = V(t_0, X_0, v_0)$$

Discretization

Since the PDE is defined continuously in the domain $[t_0,T] \times [0,\infty[\times [v_{min},v_{max}]$, it is necessary to discretize such domain in order to process the PDE numerically.

A cube grid of points will be prepared first:

$$\{t_0, t_1, \dots, t_{n-1}, t_n\} \times \{X_1, X_2, \dots, X_{m-1}, X_m\} \times \{v_1, v_2, \dots, v_{l-1}, v_l\}$$

Time axis discretization

The time points $\{t_i\}_{i=0,\dots,n}$ will be selected following these rules:

- $t0$ will match the valuation date.
- tn will match the expiry date of the option being valuated, T.
- $t0<t1\dots<tn-1<tn$.
- For each relevant date in the option (i.e., a fixing date, or a singular exercise date), such date will be part of the set $\{t_i\}_{i=0,\dots,n}$.
- The number of elements in the set will be at least the number indicated by the model parameter XX_FD_MIN_STEPS.
- The distance between two successive time steps, t_i and t_{i+1}, will not be bigger than the number of days indicated by the model parameter TV_FD_MAX_STEP_DAYS.
- The set will have as few elements as possible, provided that the above rules are respected.

Spot axis discretization

The spot points $\{X_j\}_{j=1,\dots,m}$ will be selected following these rules:

- $X1<X2\ldots<Xm-1<Xm$
- $X1\leq K\alpha$
- $Xm\geq K1-\alpha$
- $X_1\leq X_0\leq X_m$, where X_0 is the current spot
- For each relevant barrier level in the option (i.e., the level of an up continuous barrier, or a down continuous barrier), such level will be part of the set $\{X_j\}_{j=1,\dots,m}$.
- The logarithmic difference between two consecutive levels will verify:

$$\log X_j - \log X_{j-1} \leq \sqrt{\frac{VVT\ /\ n}{\text{XX\_FD\_DV2T\_TO\_DX2\_RATIO}}}$$

- The set will have as few elements as possible, provided that the above rules are respected.
- The distance between consecutive elements in the set $\{\log X_j\}_{j=1,\dots,m}$ will be as regular as possible, provided that the above rules are respected.

where:

- $\overline{VVT}$ is the average variance for the period $[t_0, T]$.
  - When Heston model is applied, the total variance can be estimated by the formula:

$$\overline{VVT} = E\left[\int_{t_0}^{T} v_s ds\right] = \int_{t_0}^{T} E[v_s] ds = \left(\theta + (v_0 - \theta)\frac{e^{-\kappa T} - 1}{-\kappa T}\right)T$$

  - When LSV model is applied, the expected variance is given by the Local Volatility. Therefore, it can be quickly approximated using equation (11.5) from 10.2 book, feed with the 25Δ Put, ATM, 25Δ $Call$ volatility points at the expiry of the option.

- K_p is the strike that verifies: $Probability(X_T \leq K_p) = p$, where the probability is calculated using the model itself. Being $P(T,K)$ the model adapted price of the put with expiry T and strike K:

$$Probability(X_T \leq K_p) = e^{\int_{t_0}^{T} r(s)ds} \cdot \frac{\partial P(T,K)}{\partial K}(K_p)$$

  The put price is known when using Heston model via Heston vanilla formulas, and when using LSV model directly from market, as the LSV model is calibrated to it.

  α has been stablished to $\alpha=0.00005=0.005\%$, which ensures that the range of the grid in the spot direction is big enough to comprise all scenarios that can have a meaningful impact in the price of the derivative.

Please note that the subindex X_0 has been avoided in the spot coordinates, to avoid confusion with the current spot. Indeed, there is no guarantee that the spot coordinates will include the current spot X_0. The most likely situation will consist of the existence of a certain index $j \in \{1,\dots,m-1\}$ such that $X_j<X_0<X_{j+1}$.

Volatility axis discretization

The volatility points $\{v_k\}_{k=1,\dots,l}$ will be selected following these rules:

- $v1<v2\ldots<vl-1<vl$

- $0 \le v_1 \le w_\alpha$. $v_1=0$ will be forced when $w_\alpha<0.00001 \cdot v_0$.
- $v_m \ge w_{1-\alpha}$
- $v_1 \le v_0 \le v_l$, where v_0 is the current volatility level
- $l \ge 3$
- The difference between two consecutive levels will verify:

$$\sqrt{v_k} - \sqrt{v_{k-1}} \le \sqrt{\frac{\overline{DDT} \,/\, (2\,n)}{\text{XX\_FD\_DV2T\_TO\_DX2\_RATIO}}}$$

- The set will have as few elements as possible, provided that the above rules are respected.
- The distance between consecutive elements in the set $\{\sqrt{v_k}\}_{k=1,\dots,l}$ will be as regular as possible, provided that the above rules are respected.

where:

- $\overline{DDT} = E\left[\int_{t_0}^{T} \frac{D(t,v_s)^2}{v_s} ds\right] = \lambda_\eta \cdot \eta^2 \cdot T$
- w_p is the volatility that verifies: $Probability(v_T \le w_p) = p$. Such probability can be computed using the known nature of v_T (it is a Cox–Ingersoll–Ross model), which distributions at future times are given by a properly parameterized Noncentral Chi-Squared distribution. More precisely:

$$Probability(v_T \le w_p) = P_{\chi^2}(w_p \cdot cs_1; cs_2, cs_3)$$

  where:

  - $P_{\chi^2}(x;k,\lambda)$ is the cumulative probability distribution function of a Noncentral Chi-Squared distribution with k degrees of freedom and component mean λ.
  - $cs_1 = \frac{4}{\lambda_\eta^2 \cdot \eta^2} \cdot \frac{-\kappa \cdot T}{\cdot e^{-k \cdot T} - 1}$
  - $cs_2 = \frac{4}{\lambda_\eta^2 \cdot \eta^2} \cdot \kappa \cdot \theta$
  - $cs_3 = \frac{4}{\lambda_\eta^2 \cdot \eta^2} \cdot \frac{-\kappa \cdot T}{\cdot e^{-k \cdot T} - 1} \cdot e^{-k \cdot T} \cdot v_0$
- α has been stablished to $\alpha = 0.00005 = 0.005\%$, which ensures that the range of the grid in the volatility direction is big enough to comprise all scenarios that can have a meaningful impact in the price of the derivative.

Please note that the subindex v_0 has been avoided in the volatility coordinates, to avoid confusion with the current volatility. Indeed, there is no guarantee that the volatility coordinates will include the current volatility v_0. The most likely situation will consist of the existence of a certain index $k \in \{1,\dots,l-1\}$ such that $v_l < v_0 < v_{l+1}$.

Discrete PDE

The algorithm will try to find a solution for the equation at the grid points. The solution will be denoted as $\{V_{ijk}\}_{i=0,\dots,n;j=1,\dots,m;k=1,\dots,l}$, being V_{ijk} the estimated solution of the equation at coordinates (t_i,X_j,v_k), i.e. $V(t_i,X_j,v_k) \approx V_{ijk}$.

As in the 1D case, the algorithm will proceed backwards, one time t_i at a time. The computation of solution at time t_i will be based entirely and exclusively in the solution at time t_{i+1} plus, perhaps, some extra boundary conditions. The equation to drive such calculation will be a discretized version of the original PDE:

- $FD_{ijk}^{t}(\{V_{ijk}\}) = \frac{V_{i+1,j,k}-V_{i,j,k}}{t_{i+1}-t_i}$
- $FD_{ijk}^{1}(\Theta,\{V_{ijk}\}) = \Theta\, V_{i,j,k} + (1-\Theta)\, V_{i+1,j,k}$
- $FD_{ijk}^{X^1}(\Theta,\{V_{ijk}\}) =$
$$\begin{cases} j=1: & \Theta\, FD_1(X_j, X_{j+1}, X_{j+2}, V_{i,j,k}, V_{i,j+1,k}, V_{i,j+2,k}, X_j) + (1-\Theta)\, FD_1(X_j, X_{j+1}, X_{j+2}, V_{i+1,j,k}, V_{i+1,j+1,k}, V_{i+1,j+2,k}, X_j) \\ 1<j<m: & \Theta\, FD_1(X_{j-1}, X_j, X_{j+1}, V_{i,j-1,k}, V_{i,j,k}, V_{i,j+1,k}, X_j) + (1-\Theta)\, FD_1(X_{j-1}, X_j, X_{j+1}, V_{i+1,j-1,k}, V_{i+1,j,k}, V_{i+1,j+1,k}, X_j) \\ j=m: & \Theta\, FD_1(X_{j-2}, X_{j-1}, X_j, V_{i,j-2,k}, V_{i,j-1,k}, V_{i,j,k}, X_j) + (1-\Theta)\, FD_1(X_{j-2}, X_{j-1}, X_j, V_{i+1,j-2,k}, V_{i+1,j-1,k}, V_{i+1,j,k}, X_j) \end{cases}$$
- $FD_{ijk}^{v^1}(\Theta,\{V_{ijk}\}) =$
$$\begin{cases} k=1: & \Theta\, FD_1(v_k, v_{k+1}, v_{k+2}, V_{i,j,k}, V_{i,j,k+1}, V_{i,j,k+2}, v_k) + (1-\Theta)\, FD_1(v_k, v_{k+1}, v_{k+2}, V_{i+1,j,k}, V_{i+1,j,k+1}, V_{i+1,j,k+2}, v_k) \\ 1<j<l: & \Theta\, FD_1(v_{k-1}, v_k, v_{k+1}, V_{i,j,k-1}, V_{i,j,k}, V_{i,j,k+1}, v_k) + (1-\Theta)\, FD_1(v_{k-1}, v_k, v_{k+1}, V_{i+1,j,k-1}, V_{i+1,j,k}, V_{i+1,j,k+1}, v_k) \\ k=l: & \Theta\, FD_1(v_{k-2}, v_{k-1}, v_k, V_{i,j,k-2}, V_{i,j,k-1}, V_{i,j,k}, v_k) + (1-\Theta)\, FD_1(v_{k-2}, v_{k-1}, v_k, V_{i+1,j,k-2}, V_{i+1,j,k-1}, V_{i+1,j,k}, v_k) \end{cases}$$
- $FD_{ijk}^{X^2}(\Theta,\{V_{ijk}\}) =$
$$\begin{cases} j=1: & \Theta\, FD_2(X_j, X_{j+1}, X_{j+2}, V_{i,j,k}, V_{i,j+1,k}, V_{i,j+2,k}) + (1-\Theta)\, FD_2(X_j, X_{j+1}, X_{j+2}, V_{i+1,j,k}, V_{i+1,j+1,k}, V_{i+1,j+2,k}) \\ 1<j<m: & \Theta\, FD_2(X_{j-1}, X_j, X_{j+1}, V_{i,j-1,k}, V_{i,j,k}, V_{i,j+1,k}) + (1-\Theta)\, FD_2(X_{j-1}, X_j, X_{j+1}, V_{i+1,j-1,k}, V_{i+1,j,k}, V_{i+1,j+1,k}) \\ j=m: & \Theta\, FD_2(X_{j-2}, X_{j-1}, X_j, V_{i,j-2,k}, V_{i,j-1,k}, V_{i,j,k}) + (1-\Theta)\, FD_2(X_{j-2}, X_{j-1}, X_j, V_{i+1,j-2,k}, V_{i+1,j-1,k}, V_{i+1,j,k}) \end{cases}$$
- $FD_{ijk}^{v^2}(\Theta,\{V_{ijk}\}) =$
$$\begin{cases} k=1: & \Theta\, FD_2(v_k, v_{k+1}, v_{k+2}, V_{i,j,k}, V_{i,j,k+1}, V_{i,j,k+2}) + (1-\Theta)\, FD_2(v_k, v_{k+1}, v_{k+2}, V_{i+1,j,k}, V_{i+1,j,k+1}, V_{i+1,j,k+2}) \\ 1<j<l: & \Theta\, FD_2(v_{k-1}, v_k, v_{k+1}, V_{i,j,k-1}, V_{i,j,k}, V_{i,j,k+1}) + (1-\Theta)\, FD_2(v_{k-1}, v_k, v_{k+1}, V_{i+1,j,k-1}, V_{i+1,j,k}, V_{i+1,j,k+1}) \\ k=l: & \Theta\, FD_2(v_{k-2}, v_{k-1}, v_k, V_{i,j,k-2}, V_{i,j,k-1}, V_{i,j,k}) + (1-\Theta)\, FD_2(v_{k-2}, v_{k-1}, v_k, V_{i+1,j,k-2}, V_{i+1,j,k-1}, V_{i+1,j,k}) \end{cases}$$
- $FD_{ijk}^{Xv}(\Theta,\{V_{ijk}\}) = \Theta\, \ddot{FD}_{\times}(i, \max(2,\min(j,m-1)), \max(2,\min(k,l-1)); X_j, v_k, \{V_{i,j,k}\}) + (1-\Theta)\, \ddot{FD}_{\times}(i, \max(2,\min(j,m-1)), \max(2,\min(k,l-1)); X_j, v_k, \{V_{i,j,k}\})$

Θ is the model parameter XX_FD_THETA controlling the particular scheme being applied.

The core finite difference formulas FD_1 and FD_2 are the same introduced in the 1D case.

$\ddot{FD}_{\times}$ is the version to estimate the cross derivative, but already linked to the grid, in order to simplify its expression:

$$\ddot{FD}_{\times}(i,j,k;x^*,y^*,\{z_{i,j,k}\}) = FD_1\Big(X_{j-1}, X_j, X_{j+1}, \begin{array}{l} FD_1(v_{k-1}, v_k, v_{k+1}, z_{i,j-1,k-1}, z_{i,j-1,k}, z_{i,j-1,k+1}, y^*), \\ FD_1(v_{k-1}, v_k, v_{k+1}, z_{i,j,k-1}, z_{i,j,k}, z_{i,j,k+1}, y^*), \\ FD_1(v_{k-1}, v_k, v_{k+1}, z_{i,j+1,k-1}, z_{i,j+1,k}, z_{i,j+1,k+1}, y^*), \quad x^* \end{array}\Big)$$

Using all the indicated formulas and properties, and going back to the discretized PDE equation for a given time index i_*, and for all $j \in \{1,\dots,m\}, k \in \{1,\dots,l\}$, there are a few realizations to make:

- There are exactly $m \cdot l$ of such equations.
- The only points participating in all those equations are exactly $\{V_{i_*,j,k}\}_{j=1,\dots,m,k=1,\dots,l} \cup \{V_{i_*+1,j}\}_{j=1,\dots,m,k=1,\dots,l}$, which is $2 \cdot m \cdot l$ points.
- If we assume known the points $\{V_{i_*+1,j}\}_{j=1,\dots,m,k=1,\dots,l}$, and unknown the points $\{V_{i_*,j,k}\}_{j=1,\dots,m,k=1,\dots,l}$, then we are left exactly with $m \cdot l$ equations and $m \cdot l$ unknowns.
- Inspecting it more closely, it can be confirmed that all the equations are linear. Indeed, the left hand side of such linear equations can be found as a linear combination of the known terms as well.

In other words, all the equations can be arranged together as:

$$A_i^I \cdot \overrightarrow{V_i} = A_i^E \cdot \overrightarrow{V_{i+1}}$$

Eq. 4

where:

- $\vec{V_i}$ is the vector $\begin{pmatrix} V_{i,1,1} \\ \vdots \\ V_{i,1,l} \\ V_{i,2,1} \\ \vdots \\ V_{i,2,l} \\ \vdots \\ \vdots \\ V_{i,m,1} \\ \vdots \\ V_{i,m,l} \end{pmatrix}$
- A_i^I, A_i^E are both $(m \cdot l) \times (m \cdot l)$ matrixes

As in the 1D case, it can be found that:

- $A_i^I = \frac{1}{\Delta t_i} I - \Theta L_i$
- $A_i^I = \frac{1}{\Delta t_i} I + (1-\Theta) L_i$

where Δt_i is simply $\Delta t_i = t_{i+1} - t_i$, I is the identity matrix of size $(m \cdot l) \times (m \cdot l)$, and L_i is an $(m \cdot l) \times (m \cdot l)$ matrix that is the same for both sides of the equation.

Finally, just applying that simplification to Eq. 4 and multiplying both sides by Δti:

$$(I - \Theta \, \Delta t_i \, L_i) \cdot \vec{V_i} = (I + (1-\Theta) \, \Delta t_i \, L_i) \cdot \overrightarrow{V_{i+1}}$$

Eq. 5

indeed:

$$\vec{V_i} = (I - \Theta \, \Delta t_i \, L_i)^{-1} \cdot (I + (1-\Theta) \, \Delta t_i \, L_i) \cdot \overrightarrow{V_{i+1}}$$

and defining the $(m \cdot l) \times (m \cdot l)$ matrix $\dddot{E}_{t_i \leftarrow t_{i+1}}$:

$$\dddot{E}_{t_i \leftarrow t_{i+1}} = (I + \Theta \, \Delta t_i \, L_i)^{-1} \cdot (I + (1-\Theta) \, \Delta t_i \, L_i)$$

the equation takes the expectation evocating form:

$$\vec{V_i} = \dddot{E}_{t_i \leftarrow t_{i+1}} \cdot \overrightarrow{V_{i+1}}$$

This is the same basic situation outlined for the 1D case, only that the matrixes defining the relationship are of size $m \cdot l$, instead of size m. In the 1D case, all the space points where calculated via a linear system, and in the 2D case, again all the space points are calculated via a linear system.

Following arguments analogous to those used in the 1D case, it is quickly found that:

- $\dddot{E}_{t_{i+1} \leftarrow t_i} \cdot \vec{U} = DF_i \cdot \vec{U} \Leftrightarrow r_i = \frac{1}{\Delta t_i} \frac{1 - DF_i}{1 - \Theta + \Theta \cdot DF_i}$
- $\dddot{E}_{t_{i+1} \leftarrow t_i} \cdot \vec{X} = DF_i^Q \cdot \vec{X} \Leftrightarrow \tilde{\mu}_i = r_i - q_i, \; q_i = \frac{1}{\Delta t_i} \frac{1 - DF_i^Q}{1 - \Theta + \Theta \cdot DF_i^Q}$

where:

- $\vec{U}$ is the $(m \cdot l) \times 1$ vector $\vec{U} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$
- $DF_i = E_{t_i \leftarrow t_{i+1}}[1](X, v) = \exp\left(-\int_{t_i}^{t_{i+1}} r(s) ds\right)$

- $\vec{X}$ is the $(m \cdot l) \times 1$ vector $\vec{X} = \begin{pmatrix} X_1 \\ \vdots \\ X_1 \\ X_2 \\ \vdots \\ X_2 \\ \vdots \\ \vdots \\ X_m \\ \vdots \\ X_m \end{pmatrix}$
- $DF_i^Q = \frac{E_{t_i \leftarrow t_{i+1}}[X](X,v)}{X} = \exp\left(-\int_{t_i}^{t_{i+1}} (r(s) - \tilde{\mu}(s)) ds\right)$

The argument can also be extended for the calculation of C_{ik}. The expectation of v is known analytically:

$$E_{t_i \leftarrow t_{i+1}}[v](X,v) = DF_i \cdot (1 - e^{-\kappa \cdot \Delta t_i}) \cdot \theta + DF_i \cdot e^{-\kappa \cdot \Delta t_i} \cdot v$$

The translation of such equality to the discrete space is:

$$\ddot{E}_{t_i \leftarrow t_{i+1}} \cdot \vec{v} = DF_i \cdot (1 - e^{-\kappa \cdot \Delta t_i}) \cdot \theta \cdot \vec{U} + DF_i \cdot e^{-\kappa \cdot \Delta t_i} \cdot \vec{v}$$

where $\vec{v}$ is the $(m \cdot l) \times 1$ vector $\vec{v} = \begin{pmatrix} v_1 \\ \vdots \\ v_l \\ v_1 \\ \vdots \\ v_l \\ \vdots \\ \vdots \\ v_1 \\ \vdots \\ v_l \end{pmatrix}$

Studying it, it is inferred that the discrete equality will hold if and only if:

$$C_{ik} = \frac{1}{\Delta t_i} \cdot \left[DF_i \cdot (1 - e^{-\kappa \cdot \Delta t_i}) \cdot \frac{1 + \Theta \cdot r_i \cdot \Delta t_i}{1 - \Theta + \Theta \cdot b_v} \cdot \theta \; + \; \left(r_i \cdot \Delta t_i - \frac{1 - b_v}{1 - \Theta + \Theta \cdot b_v} \right) \cdot v_k \right]$$

where $b_v = DF_i \cdot e^{-\kappa \cdot \Delta t_i}$.

Regarding the other coefficients of the discretized equation, those affecting second order derivatives, they will be simply approximated by their continuous coefficient counterpart. So:

- $A_{ij} = A(t_i, X_j)$
- $B_{ik} = B(t_i, v_k) = \sqrt{v_k}$
- $D_{ik} = D(t_i, v_k) = \lambda_\eta \cdot \eta \cdot \sqrt{v_k}$
- $\rho_i = \rho(t_i) = \lambda_\rho \cdot \rho$

Numerical routine

The only thing that is left in order to be able to compute the solution of the discretized PDE, is to execute each of the steps:

$$\vec{V_i} = \dddot{E}_{t_i \leftarrow t_{i+1}} \cdot \overrightarrow{V_{i+1}}$$
$$\Updownarrow$$
$$(I - \Theta\,\Delta t_i\, L_i) \cdot \vec{V_i} = (I + (1-\Theta)\,\Delta t_i\, L_i) \cdot \overrightarrow{V_{i+1}}$$

We can divide the implied computation in two sub-steps:

- Step 1: explicit step: $\overrightarrow{V_{i+\frac{1}{2}}} := (I + (1-\Theta)\,\Delta t_i\, L_i) \cdot \overrightarrow{V_{i+1}}$
- Step 2: implicit step: $\vec{V_i} = (I - \Theta\,\Delta t_i\, L_i)^{-1} \cdot \overrightarrow{V_{i+\frac{1}{2}}}$

Since $(m{\cdot}l)\times(m{\cdot}l)$ matrixes are involved in both steps, in principle this is a task with a computational cost of $O(m_3{\cdot}l3)$.

However, an even closer look at the matrix L_i, will reveal that every row will have a maximum of 9 non-zero elements. This means that the explicit computation will be of order $O(m{\cdot}l)$. Taking into account that $m{\cdot}l$ is the number of points being computed in the step, such algorithmic complexity looks unbeatable.

The situation with the implicit step, is not so favorable. Looking at the matrix L_i, it can be regarded as a tridiagonal block tridiagonal matrix (i.e., a block matrix that is tridiagonal in its block view, and with every non-zero block being tridiagonal internally). Aside from minor deviations in the edges due to boundary conditions, this allows to apply the tridiagonal matrix resolution algorithm recursively. The cost of such strategy is $O(\mathrm{m}{\cdot}l{\cdot}\min(m,l)_2)$, which is $\max\,(m,l)2$ times faster than a rough algorithm, and for moderate sizes of $\min(m,l)$ is acceptable.

However, for larger sizes of $\min(m,l)$ the algorithmic complexity becomes a serious performance problem. The technique used to overcome it, is the splitting of the matrix $(I-\Theta\,\Delta ti\; Li)$:

$$L_i = L_i^X + L_i^v + L_i^{Xv}$$

where:

- L_i^X is a special version of L_i constructed after this simplified equation:

$$-FD_{ijk}^t(\{V_{ijk}\}) \quad = \quad \frac{1}{2} A_{ij}^2\, B_{ik}^2\, X_j^2\, FD_{ij}^{X^2}(\Theta,\{V_{ijk}\}) \;+\; \tilde{\mu}_i\, X_j\, FD_{ijk}^{X^1}(\Theta,\{V_{ijk}\}) \;-\; r_i\, FD_{ijk}^1(\Theta,\{V_{ijk}\}).$$

- L_i^v is a special version of L_i constructed after this simplified equation:

$$-FD_{ijk}^t(\{V_{ijk}\}) \quad = \quad \frac{1}{2} D_{ik}^2\, FD_{ij}^{X^2}(\Theta,\{V_{ijk}\}) \;+\; C_{ik}\, FD_{ijk}^{v^1}(\Theta,\{V_{ijk}\})$$

- L_i^{Xv} is a special version of L_i constructed after this simplified equation:

$$-FD_{ijk}^t(\{V_{ijk}\}) \quad = \quad \rho_i\, A_{ij}\, B_{ik}\, D_{ik}\, X_j\, FD_{ijk}^{Xv}(\Theta,\{V_{ijk}\})$$

This decomposition verifies the desired property $L_i{=}L_{iX}{+}L_{iv}{+}L_{iXv}$. It also has the following properties:

- Liv is a block diagonal matrix, each block (almost) tridiagonal.
- LiX, after an easy reordering of its rows, is a block diagonal matrix, each block (almost) tridiagonal.

This way, the implicit step can be reorganized:

$$\vec{V}_\imath = (I - \Theta\,\Delta t_i\, L_i)^{-1} \cdot \overrightarrow{V_{\imath+\frac{1}{2}}}$$

$$\Updownarrow$$

$$(I - \Theta\,\Delta t_i\, L_i) \cdot \vec{V}_\imath = \overrightarrow{V_{\imath+\frac{1}{2}}}$$

$$\Updownarrow$$

$$(I - \Theta\,\Delta t_i\, L_i^X - \Theta\,\Delta t_i\, L_i^v - \Theta\,\Delta t_i\, L_i^{Xv}) \cdot \vec{V}_\imath = \overrightarrow{V_{\imath+\frac{1}{2}}}$$

$$\Updownarrow$$

$$(I - \Theta\,\Delta t_i\, L_i^X - \Theta\,\Delta t_i\, L_i^v) \cdot \vec{V}_\imath = \overrightarrow{V_{\imath+\frac{1}{2}}} + \Theta\,\Delta t_i\, L_i^{Xv} \cdot \vec{V}_\imath$$

$$\Updownarrow$$

$$\left(I - \Theta\,\Delta t_i\, L_i^X - \Theta\,\Delta t_i\, L_i^v + \Theta^2 \Delta {t_i}^2 L_i^v L_i^X\right) \cdot \vec{V}_\imath = \overrightarrow{V_{\imath+\frac{1}{2}}} + \Theta\,\Delta t_i\, L_i^{Xv} \cdot \vec{V}_\imath + \Theta^2 \Delta {t_i}^2 L_i^v L_i^X \cdot \vec{V}_\imath$$

$$\Updownarrow$$

$$(I - \Theta\,\Delta t_i\, L_i^v) \cdot (I - \Theta\,\Delta t_i\, L_i^X) \cdot \vec{V}_\imath = \overrightarrow{V_{\imath+\frac{1}{2}}} + \Theta\,\Delta t_i\, L_i^{Xv} \cdot \vec{V}_\imath + \Theta^2 \Delta {t_i}^2 L_i^v L_i^X \cdot \vec{V}_\imath$$

$$\Updownarrow$$

$$\vec{V}_\imath = (I - \Theta\,\Delta t_i\, L_i^X)^{-1} \cdot (I - \Theta\,\Delta t_i\, L_i^v)^{-1} \cdot \left(\overrightarrow{V_{\imath+\frac{1}{2}}} + \Theta\,\Delta t_i\, L_i^{Xv} \cdot \vec{V}_\imath + \Theta^2 \Delta {t_i}^2 L_i^v L_i^X \cdot \vec{V}_\imath\right)$$

This new arrangement of the calculations in the implicit step has the very important advantage that the linear system resolutions implied by $(I - \Theta\,\Delta t_i\, L_i^v)^{-1}$ and $(I - \Theta\,\Delta t_i\, L_i^X)^{-1}$, thanks to the tridiagonal block diagonal nature of each matrix, can both done in $O(m \cdot l)$.

The disadvantage is that the term $\left(\overrightarrow{V_{\imath+\frac{1}{2}}} + \Theta\,\Delta t_i\, L_i^{Xv} \cdot \vec{V}_\imath + \Theta^2 \Delta {t_i}^2 L_i^v L_i^X \cdot \vec{V}_\imath\right)$ requires the previous knowledge of $\vec{V}_\imath$, creating the need of knowing $\vec{V}_\imath$ in order to calculate $\vec{V}_\imath$ itself.

The dependency is broken using the Predictor-Corrector technique. The final algorithm for the calculation of $\vec{V}_\imath$ is:

1. $\vec{V}_{\imath_0} = \overrightarrow{V_{\imath+1}}$
2. Iterate: $\vec{V}_{\imath_c} = (I - \Theta\,\Delta t_i\, L_i^X)^{-1} \cdot (I - \Theta\,\Delta t_i\, L_i^v)^{-1} \cdot \left(\overrightarrow{V_{\imath+\frac{1}{2}}} + \Theta\,\Delta t_i\, L_i^{Xv} \cdot \vec{V}_{\imath_{c-1}} + \phi \cdot \Theta^2 \Delta {t_i}^2 L_i^v L_i^X \cdot \vec{V}_{\imath_{c-1}}\right)$
3. Stop when $c = \mathrm{XX\_FD\_SPLIT\_PRED\_CORR\_ITERS} + 1$.

After the predictor-corrector iteration, $\vec{V}_\imath$ is assigned the result of the last iteration: $\vec{V}_\imath \cong \vec{V}_{\imath\,\mathrm{XX\_FD\_SPLIT\_PRED\_CORR\_ITERS}+1}$

The parameter ϕ is meant to take either the value 0 or 1. Most literature implicitly set it to 0, because the term guarded by it has no impact in the theoretical convergence order of the algorithm. Also, in the special (and common) case when ρ_i=0, setting ϕ=0 makes predictor-corrector iteration unnecessary. On the other hand, setting it to 1 helps obtaining results a little closer to what the original method would yield, and would help to reduce the impact in valuation of toggling to alternative resolution techniques in the future.

ϕ is governed by the model parameter XX_FD_SPLIT_IMITATE_3RD_ORDER:

$$\phi = \begin{cases} 0 \ if \ \mathrm{XX\_FD\_SPLIT\_IMITATE\_3RD\_ORDER} \text{ is false} \\ 1 \ if \ \mathrm{XX\_FD\_SPLIT\_IMITATE\_3RD\_ORDER} \text{ is true} \end{cases}$$

Computation of A_{ij}

The computation of A_{ij} is trivial under the Heston model, simply setting A_{ij}=1, following after the continuous counterpart $A_{LV}(t,K)$=1.

In the case of the LSV model, as stated in the introduction, the relationship used to determine the continuous version is:

$$A_{LV}(t,K)^2 = \frac{\sigma_{LV}{}^2(t,K)}{E[B_{SV}^2(t,v_t)|X_t = K]}$$

In principle, when trying to assign a value to $Aij2$, the numerator $\sigma LV2(ti,Xj)$ can be replaced by:

$$\sigma_{LV}{}^2(t,X_j) \approx \sigma_{ij}^2 = \frac{1}{\Delta t_i}\int_{t_i}^{t_{i+1}} \sigma_{LV}(s,X_j)^2 ds$$

The numerator is not so trivial, but there are still some natural manipulations:

$$E\left[B_{SV}^2(t_i,v_t)|X_t = X_j\right] = \frac{\int_{v_{min}}^{v_{max}} p\left(t_0,X_0,v_0;t_i,X_j,v\right) B^2(t_i,v)\, dv}{\int_{v_{min}}^{v_{max}} p\left(t_0,X_0,v_0;t_i,X_j,v\right) dv}$$

where $p(t0,X0,v0;T,X,v)$ is the unconstrained probability of going from space point (X_0,v_0) at time t_0 to point (X_T,v_T) at time T.

If we discretize this, it is natural to start by replacing continuous probabilities by discrete ones:

$$E\left[B_{SV}^2(t_i,v_t)|X_t = X_j\right] = \frac{\sum_{k=0}^{l} \ddot{p}_{ijk}\, B^2(t_i,v_k)}{\sum_{k=0}^{l} \ddot{p}_{ijk}}$$

where $\ddot{p}_{ijk}$ is the probability of reaching the space point (X_j,v_k) at time t_i implied by our discretized PDE.

The way to calculate $\ddot{p}_{ijk}$ using brute force, would be:

$$\ddot{p}_{ijk} = P_0 \dddot{E}_{t_0 \leftarrow t_1} \dddot{E}_{t_i \leftarrow t_2} \cdots \dddot{E}_{t_{i-2} \leftarrow t_{i-1}} \dddot{E}_{t_{i-1} \leftarrow t_i} \vec{U}_{jk}$$

where:

- U_{jk} is the $m{\cdot}l$ matrix with 0 in all its positions except in the one corresponding to the grid point (X_{j,v_k}) (i.e., the element with index $j{\cdot}l+k$).
- $P0$ is the $m{\cdot}l{\times}1$ matrix that converts the grid value at time zero (the one normally represented as $\vec{V}_0$) into the result at the initial point (X_0,v_0). When $X_0=X_{j*}$ and $v_0=v_{k*}$, then naturally $P0=\vec{U}_{j*k*T}$. In other situations, $P0$ is derived from the weights implied by suitable interpolation routine.

This approach has two problems:

- Computing all the $\ddot{p}_{ijk}$, even using the splitting technique, would have a cost of $O(n_2{\cdot}m_2{\cdot}l_2)$. The cost raises to $O(n_2{\cdot}m_2{\cdot}l_2{\cdot}\min(m,l)_2)$ if using the full, non-split method.
- $\dddot{E}ti{\leftarrow}ti{+}1$ won't be completely determined until A_{ij2} is known, which is precisely the problem that it is being solved by this mean.

The first problem is simply solved by observing that just defining:

$$P_i = P_0 \dddot{E}_{t_0 \leftarrow t_1} \dddot{E}_{t_i \leftarrow t_2} \cdots \dddot{E}_{t_{i-2} \leftarrow t_{i-1}} \dddot{E}_{t_{i-1} \leftarrow t_i}$$

then it is apparent that P_i will contain, precisely, all the $\ddot{p}_{ijk}$.

$$\ddot{p}_{ijk} = P_i \cdot \vec{U}_{jk}$$

Therefore, the total operation count to compute all P_i is $O(n \cdot m \cdot l)$ if the splitting technique is used, and $O(n \cdot m \cdot l \cdot \min(m,l)^2)$ if no splitting is applied. This is approximately the same cost as computing the expectation of a future value, which is assumed to be acceptable already.

Regarding the dependency of this algorithm for discovering A_{ij}^2 precisely on A_{ij}^2, a very pragmatic approach is used: when computing the A_{ij}^2 for the period $t_i \rightarrow t_{i+1}$, only P_i is used in the summations, and P_{i+1} is not used at all. So the algorithm can proceed one step at a time:

$$P_0 \rightarrow \{A_{0j}^2\}_{j=1,\dots,m} \rightarrow P_1 \rightarrow \{A_{1j}^2\}_{j=1,\dots,m} \rightarrow P_2 \rightarrow \cdots \rightarrow P_{n-2} \rightarrow \{A_{n-2,j}^2\}_{j=1,\dots,m} \rightarrow P_{n-1} \rightarrow \{A_{n-1,j}^2\}_{j=1,\dots,m}$$

And just to state the final formula for A_{ij}^2:

$$A_{ij}^2 = \frac{1}{\Delta t_i} \frac{\left(\sum_{k=0}^{l} \ddot{p}_{ijk}\right) \int_{t_i}^{t_{i+1}} \sigma_{LV}\left(s, X_j\right)^2 ds}{\sum_{k=0}^{l} \ddot{p}_{ijk} \, B^2(t_i, v_k)}$$

Section 13. PricerFXOCarrLee

Pricer FXOCarrLee is meant to cover the most relevant cases of Calypso realized variance FX derivatives.

All Calypso realized variance derivatives can be easily valuated using Monte Carlo engines. Indeed, pricers FXOTheoreticalMonteCarlo and FXOLocalVolatilityMonteCarlo provide support to such exotics.

However, due to the high frequency of fixings typically observed in such products, a Monte Carlo engine will take relatively long to valuate them.

There are fast analytical formulas and proxies under the Black & Scholes model, available under pricer FXOTheoretical. However, the Black & Scholes model has very basic assumptions that fail to incorporate the volatility dynamics, which have a deep impact in the fair value of these products.

FXOCarrLee is provided as a model that is both smile aware and relatively fast. It can be deployed in most relevant cases of realized variance derivatives.

FXOCarrLee casts the same analytics underlying the cross-asset pricer PricerCarrLeeVolatilityDerivative, only that adapted to work seamlessly in an FXO specialized pricing context.

13.1 Pricing Parameters

Pricing Parameter	Description	Typical Value
CL_RELATIVE_TOLERANCE	**Double** It controls the accuracy of the quadrature phase in the model. The smaller the value, the higher the accuracy. It must be greater than 0. A small value is recommended to ensure that sensitivities are stable.	0.000001
CL_CONDITIONAL_VAR_SWAP_MODEL	**Domain** Conditional Variance Swaps, can't be valuated under the current FXOCarrLee pricer. Since Calypso pricer configuration scheme doesn't allow to configure Conditional Variance Swaps separately from other more Carr Lee friendly types, this escape parameter has been prepared. The Conditional Variance Swaps will be valued using the model hinted by this parameter. Possible values are (*):	Best

Pricing Parameter	Description	Typical Value
	- Theoretical / TV: valuate Conditional Variance Swaps using PricerFXOTheoretical. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate Conditional Variance Swaps using PricerFXOLocalVolatilityMonteCarlo. - Best: valuate Conditional Variance Swaps using the best model available for it (currently pointing to PricerFXOLocalVolatilityMonteCarlo). (*) Filling this pricer parameter is mandatory in the case that Conditional Variance Swaps are going to be valuated with pricer FXOCarrLee. The intention is to ensure that the user is aware of the limitations of the FXOCarrLee model before escaping to alternative valuation.	
CL_CONDITIONAL_VOL_SWAP_MODEL	**Domain** Conditional Volatility Swaps, can't be valuated under the current FXOCarrLee pricer. Since Calypso pricer configuration scheme doesn't allow to configure Conditional Volatility Swaps separately from other more Carr Lee friendly types, this escape parameter has been prepared. The Conditional Volatility Swaps will be valued using the model hinted by this parameter. Possible values are (*): - TheoreticalMonteCarlo / TVMC: valuate Conditional Volatility Swaps using PricerFXOTheoreticalMonteCarlo. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate Conditional Volatility Swaps using PricerFXOLocalVolatilityMonteCarlo. - Best: valuate Conditional Volatility Swaps using the best model available for it (currently pointing to PricerFXOLocalVolatilityMonteCarlo). (*) Filling this pricer parameter is mandatory in the case that Conditional Volatility Swaps are going to be valuated with pricer FXOCarrLee. The intention is to ensure that the user is aware of the limitations of the FXOCarrLee model before escaping to alternative valuation.	Best

Pricing Parameter	Description	Typical Value
CL_FWD_START_VOL_SWAP_MODEL	**Domain** Volatility Swaps, can't be valuated under the current FXOCarrLee pricer when their first fixing is in the future. Since Calypso pricer configuration scheme doesn't allow to configure Fwd Start Swaps separately from other more Carr Lee friendly types, this escape parameter has been prepared. The Fwd Start Volatility Swaps will be valued using the model hinted by this parameter. Possible values are (*): Theoretical / TV: valuate Fwd Start Volatility Swaps using PricerFXOTheoretical. - TheoreticalMonteCarlo / TVMC: valuate Fwd Start Volatility Swaps using PricerFXOTheoreticalMonteCarlo. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate Fwd Start Volatility Swaps using PricerFXOLocalVolatilityMonteCarlo. - Best: valuate Fwd Start Volatility Swaps using the best model available for it (currently pointing to PricerFXOLocalVolatilityMonteCarlo). (*) Filling this pricer parameter is mandatory in the case that Fwd Start Volatility Swaps are going to be valuated with pricer FXOCarrLee. The intention is to ensure that the user is aware of the limitations of the FXOCarrLee model before escaping to alternative valuation.	Best
CL_FWD_START_VAR_OPTION_MODEL	**Domain** Variance Options, can't be valuated under the current FXOCarrLee pricer when their first fixing is in the future. Since Calypso pricer configuration scheme doesn't allow to configure Fwd Start Variance Options separately from other more Carr Lee friendly types, this escape parameter has been prepared. The Fwd Start Variance Options will be valued using the model hinted by this parameter.	Best

Pricing Parameter	Description	Typical Value
	Please note that Fwd Start Variance Swaps with a cap, are internally structured as a combination of a Fwd Start Variance Swap, plus an appropriate Fwd Start Variance Option,and therefore will be valuated with themodel hinted by this parameter as well. Possible values are (*): - Theoretical / TV: valuate Fwd Start Variance Options using PricerFXOTheoretical. - TheoreticalMonteCarlo / TVMC: valuate Fwd Start Variance Options using PricerFXOTheoreticalMonteCarlo. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate Fwd Start Variance Options using PricerFXOLocalVolatilityMonteCarlo. Best: valuate Fwd Start Variance Options using the best model available for it (currently pointing to PricerFXOLocalVolatilityMonteCarlo). (*) Filling this pricer parameter is mandatory in the case that Fwd Start Variance Options are going to be valuated with pricer FXOCarrLee. The intention is to ensure that the user is aware of the limitations of the FXOCarrLee model before escaping to alternative valuation.	
CL_FWD_START_VOL_OPTION_MODEL	**Domain** Volatility Options, can't be valuated under the current FXOCarrLee pricer when their first fixing is in the future. Since Calypso pricer configuration scheme doesn't allow to configure Fwd Start Volatility Options separately from other more Carr Lee friendly types, this escape parameter has been prepared. The Fwd Start Volatility Options will be valued using the model hinted by this parameter. Please note that Fwd Start Volatility Swaps with a cap, are internally structured as a combination of a Fwd Start Volatility Swap, plus an appropriate Fwd Start Volatility Option, and therefore will be valuated with the model hinted by this parameter as well.	Best

Pricing Parameter	Description	Typical Value
	Possible values are (*): - Theoretical / TV: valuate Fwd Start Volatility Options using PricerFXOTheoretical. - TheoreticalMonteCarlo / TVMC: valuate Fwd Start Volatility Options using PricerFXOTheoreticalMonteCarlo. - LocalVolatilityMonteCarlo / Local Vol Monte Carlo / LVMC: valuate Fwd Start Volatility Options using PricerFXOLocalVolatilityMonteCarlo. - Best: valuate Fwd Start Volatility Options using the best model available for it (currently pointing to PricerFXOLocalVolatilityMonteCarlo). (*) Filling this pricer parameter is mandatory in the case that Fwd Start Volatility Options are going to be valuated with pricer FXOCarrLee. The intention is to ensure that the user is aware of the limitations of the FXOCarrLee model before escaping to alternative valuation.	
CL_THIRD_CCY_QUANTO_MODEL	**Domain** Pricer FXOCarrLee has not been extended to handle Quanto swaps (or options) yet. Since Calypso pricer configuration scheme doesn't allow to configure quanto swaps separately from other more Carr-Lee friendly types, this escape parameter has been prepared. The quanto swaps will be valued using the model hinted by this parameter. Possible values are (*): - Theoretical / TV: valuate quanto swaps using PricerFXOTheoretical. - Best: valuate quanto swaps using the best model available for it (which is PricerFXOTheoretical nowadays). (*) Even though there is only one possible value, filling this pricer parameter is mandatory in the case that quanto swaps are going to be valuated. The intention is to ensure that the user is aware of the limitations of the FXOCarrLee model before escaping to alternative TV valuation.	Best

Pricing Parameter	Description	Typical Value
	(**) The other model escape parameters will be checked first. In the event of having an assignment different from Carr-Lee model, the assigned model will be used to the evaluation of quanto expiry barriers, regardless of the value carried by CL_THIRD_CCY_QUANTO_MODEL.	

13.2 References

Carr, P. & Lee, R. Realized volatility and variance: options via swaps. Risk May 2007, pp. 76-83.

Carr, P. & Lee, R. Robust Replication of Volatility Derivatives. Preprint May 31, 2009.

Le Floch, Fabien. Volatility Derivatives Practical Notes. Available at SSRN
http://ssrn.com/abstract=2620166

13.3 Implementation details

The analytics are the same used internally by generic Calypso pricer PricerCarrLeeVolatilityDerivative. The pricerFXOCarrLee recasts such analytics for computing FX Variance Swaps following the usual FX conventions (specialmeasures, special pricing parameters, etc.).

13.4 Current limitations

This model is constrained to work on Variance Swaps, plain Volatility Swaps, plain Variance Options, and plain Volatility Options. The exact coverage is indicated in sections 2.11 and 2.12. While the coverage is quite small when compared with the wider Calypso product coverage for realized variance derivatives, it is understood to besufficient for most practical cases.

One theoretical limitation is that this model assumes continuous monitoring of the spot. In practice, the spot is onlyobserved on a discrete set of dates. The impact of this limitation is not very high if the spot is observed with an elevated frequency, like daily.

One more conceptual limitation, is that when applied to derivatives other than plain Variance Swaps, the Carr-Lee replication strategy is quite effective to capture the slope of the volatility surface, but not so much its curvature. InFX word, it is better at capturing the Risk Reversal effect, but not so good at capturing the Butterfly effect.

Another technical limitation is that this model does not take into account term structure of rates or volatilities. Thiseffect is bigger when the realized variance is computed without subtracting the mean, as is the case of the Calypsoproducts.

The pricer FXOCarrLee does not support valuation of 3rd ccy quanto options.

Section 14. Shared Pricing Parameters

There are a few Pricing Parameters that are not specific to a model implementation, but rather tied to the way inwhich market data is generated, or results are adjusted for later display.

Such parameters work across all the pricers, and are listed in the next table.

Pricing Parameter	Description	Typical Value
FX_POINTS	**Boolean** See FX & Money Markets Analytics guide.	true
USE_SMILE_VOL	**Boolean** It's normally unset, or set to true. Set it to false to force the current pricer to ignore the smile information from the volatility surface, and work with ATM Volatilities only.	true
FX_VOL_SMILE_INTERPOLATION	**String** It's normally unset, or set to Default. If set differently, it enforces a particular methodology for interpolating volatilities, which overrides general Calypso settings. Possible values: Default: as set up in the corresponding FX Volatility Surface configured in the Pricing Environment. Natural Quintic Spline v1: for every date, natural quantic spline of $\sigma 4$ against log K is applied to interpolate among the pillar points. Extrapolation is done via continuing parabola in the same log $K - \sigma 4$ space, except amends done when necessary to keep the curve positive.	Default
ZD_PRICING	**Boolean** See FX & Money Markets Analytics guide.	false
INCLUDE_FEES	**Boolean** See FX & Money Markets Analytics guide.	
NPV_INCLUDE_CASH	**Boolean** See FX & Money Markets Analytics guide.	

Legacy Parameters

There are some hidden parameters that Calib pricers react to. Those parameters are not officially supported by Calib FXO pricers, neither are they regular, generic, Calypso Pricing Parameters. However, certain Calypso toolsexpected them to be supported by every FX option pricer in former Calypso versions, and in order to simplify deployment of Calypso pricers, it was decided to integrate them.

Some of those parameters are listed here, just in case it can help identifying unexpected behavior coming out ofFXO pricers:

- **USE_ATM_VOL**. Regular parameter of classic pricers. Calib pricers will ignore it if unset or set to false, but will respect it if set to true, even in the case when USE_SMILE_VOL is set to false.
- **USE_ATM_VOL_VA**. Regular parameter of classic pricers. Calib pricers will ignore it if unset or set to false, but will make an attempt to respect it if set to true, even in the case when USE_SMILE_VOL is set to false.
- **LEGACY_VOL_LOOKUP**. This parameter controls how interpolation of volatilities is performed when FX Volatility Surface generator is FXOptionDelta. For pricers to react to it, it has to be configured as a Model Parameter of classic pricer PricerFXOptionVanilla. If set to false, Calib pricers will perform interpolation over FXOptionDelta generated surfaces, same way as it is done for FX Option generated surfaces. If not set, or set to true, then interpolation over FXOptionDelta generated surfaces will be performed accordingly to that generator's original behavior. Please refer to specific documentation on this parameter to obtain a better understanding of its implications.

Section 15. Recommended Pricer Configuration

FXO pricers are configurable so that their behavior can be customized to relevant needs.

Next setup is a proposal based upon direct knowledge of different pricers advantages and disadvantages, as well asthe feedback received while validating the analytics with different Calypso customers.

Since Pricer Parameters are configured at different levels, the proposal shows how to configure them at each level.The first level is always necessary to just have the parameters available in Calypso. Then, it is recommended that at least one of the other two levels is filled in order to set up the preferred values. In general, just configuring parameters in the first two levels will be the most effective approach.

Finally, the last sub-section shows how a recommended assignment of pricers to different sub-types of FX Options.

15.1 Configuration > System > Add Pricer

Product	Pricer
FXOption	PricerFXOTheoretical
FXOption	PricerFXOMarket
FXOption	PricerFXOVannaVolga
FXOption	PricerFXOHeston
FXOption	PricerFXOTheoreticalMonteCarlo
FXOption	PricerFXOLocalVolatilityMonteCarlo
FXOption	PricerFXOTheoreticalFiniteDifference
FXOption	PricerFXOLocalVolatilityFiniteDifference
FXOption	PricerFXOHestonFiniteDifference
FXOption	PricerFXOLocalStochasticVolatilityFiniteDifference
VarianceSwap	PricerFXOTheoretical
VarianceSwap	PricerFXOCarrLee

Product	Pricer
VarianceSwap	PricerFXOTheoreticalMonteCarlo
VarianceSwap	PricerFXOLocalVolatilityMonteCarlo
VarianceOption	PricerFXOTheoretical
VarianceOption	PricerFXOCarrLee
VarianceOption	PricerFXOTheoreticalMonteCarlo
VarianceOption	PricerFXOLocalVolatilityMonteCarlo

15.2 Configuration > System > Add Pricer Parameter Type

Please take into account that even though the Default Value indicated here can be consumed by different parts ofthe system, they will generally not be referenced by the FXO pricers directly.

Pricing Param Name	Type	Domain
TV_ATM_TYPE	String	Market Concordant,Forward
TV_USE_FLAT_TERM_STRUCTURE	Boolean	true,false
TV_ASIAN_ARITH_PROXY	String	Log Normal,Shifted Log Normal
TV_CONDITIONAL_VOL_SWAP_MODEL	String	TheoreticalMonteCarlo,Best
MKT_SELF_QUANTO_MODEL	String	Theoretical,VannaVolga,LocalVolatilityMonteCarlo, Best
MKT_THIRD_CCY_QUANTO_MODEL	String	Theoretical
MKT_ACCRUAL_FADER_MODEL	String	Theoretical,VannaVolga,LocalVolatilityMonteCarlo,LocalVolatilityFiniteDifference,LocalStochasticVolatilityFiniteDifference,Best
MKT_ASIAN_CASH_IN_PRIMARY_MODEL	String	Theoretical,LocalVolatilityMonteCarlo,Best
VV_REFERENCE_DELTA	Double	

Pricing Param Name	Type	Domain
VV_REFERENCE_DELTA_TYPE	String	Market Concordant,Symmetric,Symmetric Proxy
VV_WEIGHT_POLICY	String	Full,Expected Life Fraction,Survival Probability,User input,Fisher
VV_INPUT_WEIGHT	Double	
VV_WEIGHT_USE_SYMMETRIC_PROB	Boolean	true,false
VV_HEDGE_POLICY	String	Synthetic,Natural
VV_HEDGE_EXPIRY	String	Barrier End Date,Expiry Date,Expected Exit Date
VV_SPARE_TERMINAL_ADJUSTMENT	Boolean	true,false
VV_TERMINAL_REF_MODEL	String	Market,VannaVolga
VV_CONSISTENCY_ENFORCEMENT	String	None,Basic,Advanced
VV_5_POINTS_ADJUSTMENT	Boolean	true,false
VV_REFERENCE_EXTRA_DELTA	Double	
VV_EXPIRY_BARRIER_MODEL	String	Market,VannaVolga
VV_THIRD_CCY_QUANTO_MODEL	String	Theoretical
VV_ACCRUAL_RESETTABLE_MODEL	String	Theoretical,Market,LocalVolatilityMonteCarlo,Best
H_REFERENCE_DELTA	Double	
H_MEAN_REVERSION_1Y	Double	
TV_MC_ITERATIONS	Integer	
TV_MC_PAYOFF_SMOOTHING_PV	Double	
TV_MC_PAYOFF_SMOOTHING_SNS	Double	

Pricing Param Name	Type	Domain
LV_MC_MAX_STEP_DAYS	Double	
LV_MC_ITERATIONS	Integer	
LV_MC_PAYOFF_SMOOTHING_PV	Double	
LV_MC_PAYOFF_SMOOTHING_SNS	Double	
LV_MC_TV_MODEL	String	Theoretical,TheoreticalMonteCarlo
TV_FD_MAX_STEP_DAYS	Double	
TV_FD_MIN_STEPS	Integer	
TV_FD_DV2T_TO_DX2_RATIO	Double	
TV_FD_THETA	Double	
TV_FD_MAX_THREADS	Integer	
LV_FD_MAX_STEP_DAYS	Double	
LV_FD_MIN_STEPS	Integer	
LV_FD_DV2T_TO_DX2_RATIO	Double	
LV_FD_THETA	Double	
LV_FD_MAX_THREADS	Integer	
LV_FD_TV_MODEL	String	Theoretical,TheoreticalFiniteDifference
H_FD_MAX_STEP_DAYS	Double	
H_FD_MIN_STEPS	Integer	
H_FD_DV2T_TO_DX2_RATIO	Double	
H_FD_THETA	Double	

Pricing Param Name	Type	Domain
H_FD_SPLIT	Boolean	
H_FD_SPLIT_PRED_CORR_ITERS	Integer	
H_FD_SPLIT_IMITATE_3RD_ORDER	Boolean	
H_FD_MAX_THREADS	Integer	
H_FD_TV_MODEL	String	Theoretical,TheoreticalFiniteDifference
LSV_STOCHASTIC_VOL_WEIGHT	Double	
LSV_CORRELATION_WEIGHT	Double	
LSV_FD_MAX_STEP_DAYS	Double	
LSV_FD_MIN_STEPS	Integer	
LSV_FD_DV2T_TO_DX2_RATIO	Double	
LSV_FD_THETA	Double	
LSV_FD_SPLIT	Boolean	
LSV_FD_SPLIT_PRED_CORR_ITERS	Integer	
LSV_FD_SPLIT_IMITATE_3RD_ORDER	Boolean	
LSV_FD_MAX_THREADS	Integer	
LSV_FD_TV_MODEL	String	Theoretical,TheoreticalFiniteDifference
LSV_EXPIRY_BARRIER_MODEL	String	Market,LocalStochasticVolatilityFiniteDifference
LSV_THIRD_CCY_QUANTO_MODEL	String	Theoretical
LSV_ACCRUAL_RESETTABLE_MODEL	String	Theoretical,Market,LocalVolatilityMonteCarlo,Best
CL_RELATIVE_TOLERANCE	Double	

Pricing Param Name	Type	Domain
CL_CONDITIONAL_VAR_SWAP_MODEL	String	Theoretical,TheoreticalMonteCarlo,LocalVolatilityMonteCarlo,Best
CL_CONDITIONAL_VOL_SWAP_MODEL	String	TheoreticalMonteCarlo,LocalVolatilityMonteCarlo,Best
CL_FWD_START_VOL_SWAP_MODEL	String	Theoretical,TheoreticalMonteCarlo,LocalVolatilityMonteCarlo,Best
CL_FWD_START_VAR_OPTION_MODEL	String	Theoretical,TheoreticalMonteCarlo,LocalVolatilityMonteCarlo,Best
CL_FWD_START_VOL_OPTION_MODEL	String	Theoretical,TheoreticalMonteCarlo,LocalVolatilityMonteCarlo,Best
CL_THIRD_CCY_QUANTO_MODEL	String	Theoretical,Best

15.3 Market Data > Pricing Environment > Pricer Parameters Set

Find highlighted the Pricing Parameters which recommended value is different from the default value (i.e., thevalue that the pricer will assume if no further information is found in any other place in the system).

To simplify the analysis, the default value is provided in an additional column.

Product Type	Name	Value	Default Value
FXOption	TV_ATM_TYPE	Market Concordant	Market Concordant
FXOption	TV_USE_FLAT_TERM_STRUCTURE	false	false
FXOption	TV_ASIAN_ARITH_PROXY	Log Normal	Log Normal
FXOption	TV_CONDITIONAL_VOL_SWAP_MODEL	Best	N/A
FXOption	MKT_SELF_QUANTO_MODEL	Best	N/A
FXOption	MKT_THIRD_CCY_QUANTO_MODEL	Theoretical	N/A
FXOption	MKT_ACCRUAL_FADER_MODEL	Best	N/A

Product Type	Name	Value	Default Value
FXOption	MKT_ASIAN_CASH_IN_PRIMARY_MODEL	Best	N/A
FXOption	VV_REFERENCE_DELTA	25	25
FXOption	VV_REFERENCE_DELTA_TYPE	Market Concordant	Market Concordant
FXOption	VV_WEIGHT_POLICY	Expected Life Fraction	Expected Life Fraction
FXOption	VV_INPUT_WEIGHT	1	1
FXOption	VV_WEIGHT_USE_SYMMETRIC_PROB	true	true
FXOption	VV_HEDGE_POLICY	Synthetic	Synthetic
FXOption	VV_HEDGE_EXPIRY	Barrier End Date	Barrier End Date
FXOption	VV_SPARE_TERMINAL_ADJUSTMENT	true	true
FXOption	VV_TERMINAL_REF_MODEL	Market	Market
FXOption	VV_CONSISTENCY_ENFORCEMENT	Basic	Basic
FXOption	VV_5_POINTS_ADJUSTMENT	false	False
FXOption	VV_REFERENCE_EXTRA_DELTA	10	10
FXOption	VV_EXPIRY_BARRIER_MODEL	Market	Market
FXOption	VV_THIRD_CCY_QUANTO_MODEL	Theoretical	N/A
FXOption	VV_ACCRUAL_RESETTABLE_MODEL	Best	N/A
FXOption	H_REFERENCE_DELTA	25	25
FXOption	H_MEAN_REVERSION_1Y	1	1
FXOption	TV_MC_ITERATIONS	32767	32767

Product Type	Name	Value	Default Value
FXOption	TV_MC_PAYOFF_SMOOTHING_PV	0	0
FXOption	TV_MC_PAYOFF_SMOOTHING_SNS	0.125	0.125
FXOption	LV_MC_MAX_STEP_DAYS	14	14
FXOption	LV_MC_ITERATIONS	32767	32767
FXOption	LV_MC_PAYOFF_SMOOTHING_PV	0	0
FXOption	LV_MC_PAYOFF_SMOOTHING_SNS	0.125	0.125
FXOption	LV_MC_TV_MODEL	Theoretical	Theoretical
FXOption	TV_FD_MAX_STEP_DAYS	7	7
FXOption	TV_FD_MIN_STEPS	75	75
FXOption	TV_FD_DV2T_TO_DX2_RATIO	0.5	0.5
FXOption	TV_FD_THETA	0.5	0.5
FXOption	TV_FD_MAX_THREADS	1	1
FXOption	LV_FD_MAX_STEP_DAYS	7	7
FXOption	LV_FD_MIN_STEPS	75	75
FXOption	LV_FD_DV2T_TO_DX2_RATIO	0.5	0.5
FXOption	LV_FD_THETA	0.5	0.5
FXOption	LV_FD_MAX_THREADS	1	1
FXOption	LV_FD_TV_MODEL	Theoretical	Theoretical
FXOption	H_FD_MAX_STEP_DAYS	7	7
FXOption	H_FD_MIN_STEPS	75	75

Product Type	Name	Value	Default Value
FXOption	H_FD_DV2T_TO_DX2_RATIO	0.5	0.5
FXOption	H_FD_THETA	0.5	0.5
FXOption	H_FD_SPLIT	true	true
FXOption	H_FD_SPLIT_PRED_CORR_ITERS	1	1
FXOption	H_FD_SPLIT_IMITATE_3RD_ORDER	true	true
FXOption	H_FD_MAX_THREADS	1	1
FXOption	H_FD_TV_MODEL	Theoretical	Theoretical
FXOption	LSV_STOCHASTIC_VOL_WEIGHT	0.6	0.6
FXOption	LSV_CORRELATION_WEIGHT	0.0	0.0
FXOption	LSV_FD_MAX_STEP_DAYS	7	7
FXOption	LSV_FD_MIN_STEPS	75	75
FXOption	LSV_FD_DV2T_TO_DX2_RATIO	0.5	0.5
FXOption	LSV_FD_THETA	0.5	0.5
FXOption	LSV_FD_SPLIT	true	true
FXOption	LSV_FD_SPLIT_PRED_CORR_ITERS	1	1
FXOption	LSV_FD_SPLIT_IMITATE_3RD_ORDE R	true	true
FXOption	LSV_FD_MAX_THREADS	1	1
FXOption	LSV_FD_TV_MODEL	Theoretical	Theoretical
FXOption	LSV_EXPIRY_BARRIER_MODEL	Market	Market
FXOption	LSV_THIRD_CCY_QUANTO_MODEL	Theoretical	N/A

Product Type	Name	Value	Default Value
FXOption	LSV_ACCRUAL_RESETTABLE_MODEL	Best	N/A
VarianceSwap	CL_RELATIVE_TOLERANCE	0.000001	0.000001
VarianceSwap	CL_CONDITIONAL_VAR_SWAP_MODEL	Best	N/A
VarianceSwap	CL_CONDITIONAL_VOL_SWAP_MODEL	Best	N/A
VarianceSwap	CL_FWD_START_VOL_SWAP_MODEL	Best	N/A
VarianceSwap	CL_FWD_START_VAR_OPTION_MODEL	Best	N/A
VarianceSwap	CL_FWD_START_VOL_OPTION_MODEL	Best	N/A
VarianceSwap	CL_THIRD_CCY_QUANTO_MODEL	Best	N/A
VarianceOption	CL_RELATIVE_TOLERANCE	0.000001	0.000001
VarianceOption	CL_FWD_START_VAR_OPTION_MODEL	Best	N/A
VarianceOption	CL_FWD_START_VOL_OPTION_MODEL	Best	N/A
VarianceOption	CL_THIRD_CCY_QUANTO_MODEL	Best	N/A

15.4 Market Data > Pricing Environment > Pricer Configuration >Model Parameters

Find highlighted the Pricing Parameters which recommended value is different from the default value (i.e., thevalue that the pricer will assume if no further information is found in any other place in the system).

To simplify the analysis, the default value is provided in an additional column.

Pricer	Pricing Param Name	Value	Default Value
PricerFXOTheoretical	TV_ATM_TYPE	Market Concordant	Market Concordant
PricerFXOTheoretical	TV_USE_FLAT_TERM_STRUCTURE	false	false

Pricer	Pricing Param Name	Value	Default Value
PricerFXOTheoretical	TV_ASIAN_ARITH_PROXY	Log Normal	Log Normal
PricerFXOTheoretical	TV_CONDITIONAL_VOL_SWAP_MODEL	Best	N/A
PricerFXOMarket	MKT_SELF_QUANTO_MODEL	Best	N/A
PricerFXOMarket	MKT_THIRD_CCY_QUANTO_MODEL	Theoretical	N/A
PricerFXOMarket	MKT_ACCRUAL_FADER_MODEL	Best	N/A
PricerFXOMarket	MKT_ASIAN_CASH_IN_PRIMARY_MODEL	Best	N/A
PricerFXOVannaVolga	VV_REFERENCE_DELTA	25	25
PricerFXOVannaVolga	VV_REFERENCE_DELTA_TYPE	Market Concordant	Market Concordant
PricerFXOVannaVolga	VV_WEIGHT_POLICY	Expected Life Fraction	Expected Life Fraction
PricerFXOVannaVolga	VV_INPUT_WEIGHT	1	1
PricerFXOVannaVolga	VV_WEIGHT_USE_SYMMETRIC_PROB	true	true
PricerFXOVannaVolga	VV_HEDGE_POLICY	Synthetic	Synthetic
PricerFXOVannaVolga	VV_HEDGE_EXPIRY	Barrier End Date	Barrier End Date
PricerFXOVannaVolga	VV_SPARE_TERMINAL_ADJUSTMENT	true	true
PricerFXOVannaVolga	VV_TERMINAL_REF_MODEL	Market	Market
PricerFXOVannaVolga	VV_CONSISTENCY_ENFORCEMENT	Basic	Basic
PricerFXOVannaVolga	VV_5_POINTS_ADJUSTMENT	false	false
PricerFXOVannaVolga	VV_REFERENCE_EXTRA_DELTA	10	10
PricerFXOVannaVolga	VV_EXPIRY_BARRIER_MODEL	Market	Market

Pricer	Pricing Param Name	Value	Default Value
PricerFXOVannaVolga	VV_THIRD_CCY_QUANTO_MODEL	Theoretical	N/A
PricerFXOVannaVolga	VV_ACCRUAL_RESETTABLE_MODEL	Best	N/A
PricerFXOTheoretical MonteCarlo	TV_ATM_TYPE	Market Concordant	Market Concordant
PricerFXOTheoretical MonteCarlo	TV_USE_FLAT_TERM_STRUCTURE	false	false
PricerFXOTheoretical MonteCarlo	TV_MC_ITERATIONS	32767	32767
PricerFXOTheoretical MonteCarlo	TV_MC_PAYOFF_SMOOTHING_PV	0	0
PricerFXOTheoretical MonteCarlo	TV_MC_PAYOFF_SMOOTHING_SNS	0.125	0.125
PricerFXOLocalVolatili tyMonteCarlo	LV_MC_MAX_STEP_DAYS	14	14
PricerFXOLocalVolatili tyMonteCarlo	LV_MC_ITERATIONS	32767	32767
PricerFXOLocalVolatili tyMonteCarlo	LV_MC_PAYOFF_SMOOTHING_PV	0	0
PricerFXOLocalVolatili tyMonteCarlo	LV_MC_PAYOFF_SMOOTHING_SNS	0.125	0.125
PricerFXOLocalVolatili tyMonteCarlo	LV_MC_TV_MODEL	Theoretical	Theoretical
PricerFXOHeston	H_REFERENCE_DELTA	25	25
PricerFXOHeston	H_MEAN_REVERSION_1Y	1	1
PricerFXOTheoretical FiniteDifference	TV_ATM_TYPE	Market Concordant	Market Concordant
PricerFXOTheoretical FiniteDifference	TV_USE_FLAT_TERM_STRUCTURE	false	false

Pricer	Pricing Param Name	Value	Default Value
PricerFXOTheoreticalFiniteDifference	TV_FD_MAX_STEP_DAYS	7	7
PricerFXOTheoreticalFiniteDifference	TV_FD_MIN_STEPS	75	75
PricerFXOTheoreticalFiniteDifference	TV_FD_DV2T_TO_DX2_RATIO	0.5	0.5
PricerFXOTheoreticalFiniteDifference	TV_FD_THETA	0.5	0.5
PricerFXOTheoreticalFiniteDifference	TV_FD_MAX_THREADS	1	1
PricerFXOLocalVolatilityFiniteDifference	LV_FD_MAX_STEP_DAYS	7	7
PricerFXOLocalVolatilityFiniteDifference	LV_FD_MIN_STEPS	75	75
PricerFXOLocalVolatilityFiniteDifference	LV_FD_DV2T_TO_DX2_RATIO	0.5	0.5
PricerFXOLocalVolatilityFiniteDifference	LV_FD_THETA	0.5	0.5
PricerFXOLocalVolatilityFiniteDifference	LV_FD_MAX_THREADS	1	1
PricerFXOLocalVolatilityFiniteDifference	LV_FD_TV_MODEL	Theoretical	Theoretical
PricerFXOHestonFiniteDifference	H_REFERENCE_DELTA	25	25
PricerFXOHestonFiniteDifference	H_MEAN_REVERSION_1Y	1	1
PricerFXOHestonFiniteDifference	H_FD_MIN_STEPS	75	75
PricerFXOHestonFiniteDifference	H_FD_DV2T_TO_DX2_RATIO	0.5	0.5

Pricer	Pricing Param Name	Value	Default Value
PricerFXOHestonFiniteDifference	H_FD_THETA	0.5	0.5
PricerFXOHestonFiniteDifference	H_FD_SPLIT	true	true
PricerFXOHestonFiniteDifference	H_FD_SPLIT_PRED_CORR_ITERS	1	1
PricerFXOHestonFiniteDifference	H_FD_SPLIT_IMITATE_3RD_ORDER	true	true
PricerFXOHestonFiniteDifference	H_FD_TV_MODEL	Theoretical	Theoretical
PricerFXOLocalStochasticVolatilityFiniteDifference	H_REFERENCE_DELTA	25	25
PricerFXOLocalStochasticVolatilityFiniteDifference	H_MEAN_REVERSION_1Y	1	1
PricerFXOLocalStochasticVolatilityFiniteDifference	LSV_STOCHASTIC_VOL_WEIGHT	0.60	0.60
PricerFXOLocalStochasticVolatilityFiniteDifference	LSV_CORRELATION_WEIGHT	0	0
PricerFXOLocalStochasticVolatilityFiniteDifference	LSV_FD_MIN_STEPS	75	75
PricerFXOLocalStochasticVolatilityFiniteDifference	LSV_FD_DV2T_TO_DX2_RATIO	0.5	0.5
PricerFXOLocalStochasticVolatilityFiniteDifference	LSV_FD_THETA	0.5	0.5
PricerFXOLocalStochasticVolatilityFiniteDifference	LSV_FD_SPLIT	true	true

Pricer	Pricing Param Name	Value	Default Value
PricerFXOLocalStochasticVolatilityFiniteDifference	LSV_FD_SPLIT_PRED_CORR_ITERS	1	1
PricerFXOLocalStochasticVolatilityFiniteDifference	LSV_FD_SPLIT_IMITATE_3RD_ORDER	true	true
PricerFXOLocalStochasticVolatilityFiniteDifference	LSV_FD_TV_MODEL	Theoretical	Theoretical
PricerFXOCarrLee	CL_RELATIVE_TOLERANCE	0.000001	0.000001
PricerFXOCarrLee	CL_CONDITIONAL_VAR_SWAP_MODEL	Best	N/A
PricerFXOCarrLee	CL_CONDITIONAL_VOL_SWAP_MODEL	Best	N/A
PricerFXOCarrLee	CL_FWD_START_VOL_SWAP_MODEL	Best	N/A
PricerFXOCarrLee	CL_FWD_START_VAR_OPTION_MODEL	Best	N/A
PricerFXOCarrLee	CL_FWD_START_VOL_OPTION_MODEL	Best	N/A
PricerFXOCarrLee	CL_THIRD_CCY_QUANTO_MODEL	Best	N/A

15.5 Market Data > Pricing Environment > Pricer Configuration > Pricer

This recommended configuration comprises all subtypes of FXOption available in Calypso.

Calib pricers are recommended over classic ones whenever available for a certain subtype.

Product	ExtendedType	SubType	Pricer
FXOption	ANY	European	PricerFXOMarket
FXOption	ANY	American	PricerFXOMarket
FXOption	ANY	FWDSTART	PricerFXOMarket or PricerFXOLocalVolatilityMonteCarlo

Product	ExtendedType	SubType	Pricer
FXOption	ANY	BARRIER	PricerFXOLocalStochasticVolatilityFiniteDifference
FXOption	ANY	DIGITAL	PricerFXOLocalStochasticVolatilityFiniteDifference
FXOption	ANY	DIGITALWITHBARRIER	PricerFXOLocalStochasticVolatilityFiniteDifference
FXOption	ANY	ACCRUAL	PricerFXOMarket or PricerFXOLocalVolatilityMonteCarlo
FXOption	ANY	ASIAN	PricerFXOMarket or PricerFXOLocalVolatilityMonteCarlo
FXOption	ANY	LOOKBACK	PricerFXOLocalVolatilityMonteCarlo
FXOption	ANY	COMPOUND	PricerFXOLocalStochasticVolatilityFiniteDifference
FXOption	ANY	VOLFWD	PricerFXOTheoretical
VarianceSwap	ANY	FX	PricerFXOCarrLee
VarianceOption	ANY	FX	PricerFXOCarrLee

Section 16. Section 16. Pricer Measures

The FXOption Pricer Measures produced by the FXO pricers, are the same that can be found in classic FXOption pricers. Please refer to fxmm_analytics guide to understand their meaning.

Regarding the implementation, it has been decided that initially, all measures will be internally calculated by a Finite Differences engine. Then, they will be adapted to externally expected conventions applying necessary transformations.

The Finite Difference implementation simplifies greatly the development of new pricers, as the focus can be put on the calculation of the price itself, and then all measures will automatically work using the same, shared Finite Difference engine. Even though using other strategies for the derivation of measures could improve accuracy and performance, those alternatives will only be considered as an enhancement on existing pricers, and only once the existing pricers have been proved to be mature and the need for newer or better pricing methodologies is reduced.

Since documentation on Finite Difference formulas is widely available, the concept will not be explained here. Further details, like whether the formulas are centered or not, or the size of the step applied internally to estimate the derivative, will not be published here directly. The main priorities when estimating derivatives using Finite Differences is that they are stable, as accurate as possible, and then as fast as possible. For that reason, depending on the context, and the model being used, different choices are done following a practical study of the model behavior. In case that changes are introduced on the existing choices, that will be indicated on this document's section Relevant Changes History.

One of the main drawbacks of Finite Difference derivatives when used on financial exotics, is their likely unstability on the vicinity of product or model boundaries, like a barrier, or a strike, or a subjective exercise triggering level. The FXO pricers make explicit efforts to avoid that problem by keeping those relevant boundaries conveniently mapped, and then making sure that all Finite Differences valuations are kept at the same side of the boundaries.

Sticky Strike measures

While measures like MOD_DELTA, MOD_VEGA, REAL_RHO or REAL_THETA follow a pre-defined way when assuming the reaction of the volatility surface to the change on parameters, most other measures need to define an explicit policy about that, and stick to it.

Current version of FXO ricers, assume a *Sticky Strike* behavior. Which is to say that, if we have a **R**eference set ofparameters, and a **S**hifted set of parameters, where all the relevant inputs (spot, interest rates) have been changed in some arbitrary way, and particularly the volatility surface is supposed to be shifted by a parameterizedshift $h_\sigma(t, K)$, then the volatility mapping for the Shifted parameters meets this relationship:

$$\bar{\sigma}_S(t_0, t, K) = \bar{\sigma}_R(t_0, t, K) + h_\sigma(t, K)$$

Or in other words, it is assumed that the existing volatility surface is given in terms of expiry and strike of an option, and that such a parameterization is stable when other market parameters change.

That is not true in practice, but in particular in the FX markets, that is the assumption done when calculating pricing measures for Vanillas. For that reason, Sticky Strike has been chosen to be the first policy to implement. Other schemes (Sticky Delta, TV greeks, or hybrid Strike/Delta schemes) will be considered for future enhancements.