GSR Development Overview

Peter Caspers

April 6, 2013

Table of contents

1. Model
2. Instruments
3. Pricing Engines
4. QuantLib extensions
5. Nonstandard Swaption Calibration
6. Summary

Model Spec

- GSR Model = Gaussian Short Rate Model = Hull White Model
- 1 factor
- time dependent deterministic volatility and reversion
- dynamics is in T-Forward measure
- formulation without "fitting function"

Model Class Gsr

- constructors for fixed mean reversion, calibrated mean reversion
- sugar constructor for constant mean reversion
- state process class GsrProcess with exact formulas
- caching necessary because of fomula complexity

Instrument Class NonStandardSwap

- fixed float swap with periodic notionals and / or fixed rates
- direct generalization of Vanilla Swap
- copy constructor from Vanilla Swap

Instrument Class NonStandardSwaption

- option on NonstandardSwap
- just like Swaption is an option on Vanilla Swap
- copy constructor from Swaption
- physical delivery
- european or bermudan exercise

Engine Class GsrSwaptionEngine

- pricing engine for european and bermudan swaptions
- static multicurve support
- uses numerical integration
- payoff is interpolated with Lagrange Splines
- exact integration of splines against gaussian distribution
- coverage of n (default 7) standard deviations of the state variable with m (default 64) grid points
- backward engine for bermudan exercise rights

Engine Class GsrJamshidianSwaptionEngine

- pricing engine for european swaptions
- uses the Jamshidian trick
- exact pricing
- monocurve only
- extended version of the ql implementation respecting the start delay of the underlying swap
- integral engine is usually just as good (accuracy, speed)

Engine Class GsrNonstandardSwaptionEngine

- pricing engine for european and bermudan swaptions with periodic notional and / or fixed rate
- static multicurve support, uses numerical integration, just as GsrSwaptionEngine
- provides generation of suitable calibration baskets (method see below), baskets can be retrieved from the instrument

SwaptionHelper

- provide non atm calibration instruments (ensuring that they are otm)
- provide nominal $\neq 1$ (meaningless for calibration, but useful information for non standard calibration baskets)

Iterative model calibration

- provide an iterative model calibration in addition to the global optimization
- for boostrapping the piecewise volatility function in the GSR model
- much faster than global calibration if many volatility step dates are present (> 10)
- of course not generally applicable, so handle with care (similar situation as with iterative and global bootstrapper for yield curves)

Representative Swaption Approach

Problem: Find a standard (market quoted up to interpolation in option maturity or swap length) swaption which is "similar" to a non standard european swaption.

- "Similar" should mean: the underlyings on which the options are written should at option expiry have close npvs in all states of the world. If this is the case the payoff from both options will be close, too, hence todays npv of both options thereby inducing the npv of the non standard option thanks to the market value of the quoted option.
- Obviously "in all states of the world" is a model dependent concept. Here we use the GSR model which use for the non standard bermudan swaption pricing later on.

European non standard swaptions

- obviously the approach can also be used to price non standard european swaptions
- note that the model choice implies perfect correlation of rates
- other models are possible ...

Procedure to find the representative swaption

- the underlying has 3 free parameters: nominal, fixed rate, maturity
- we determine these parameters by minimizing the difference in npv, delta and gamma of the two underlyings
- delta and gamma should be understood as first and second order derivatives of the npv by the state variable of the model
- npv, delta and gamma are evaluated at the expectation of the state variable at option expiry

Details of the procedure

- the minimization is done using numerical optimization (Levenberg Marquardt)
- swap maturity is discrete in nature (we use whole months as unit). Since optimization (and in particular Levenberg Marquardt) relies on continuous / differentiable target functions we interpolate npvs between these "physical" swaps
- furthermore to give the optimizer a nice target function, negative nominals and negative fixed rate are interpreted as payer / receiver swaps
- delta and gamma are computed by finite differences and expressed as derivatives in the normalized (zero mean, unit variance) state variable
- npv and delta are normalized by the target delta, gamma by the target gamma

A fixed point problem ?

- the npv, delta and gamma depend on the model parameters (sigma up to expiry, reversion up to forward measure time)
- therefore after calibration of the model to the basket, the basket changes !
- should we iterate to find some fixed point ?
- no, because the dependency of the basket on the model parameters is weak enough to determine the basket on reasonable model parameters and then calibrating the model the that basket. full stop.

Calibration Baskets: Demo / Test cases

- standard swaption
- decreasing notional (amortizer)
- increasing notional (accreter)
- step up coupon
- step down coupon
- decreasing notional and step up coupon

Robustness of the representation: Demo

- match a standard swap
- plot the npv of the non standard and the standard swap in 5 standard deviations of the state variable
- price the standard swaption and the non standard swaption

Summary

- Gsr model class provides time dependent volatility and reversion
- pricing engines provide static multi curve support
- extended non atm swaption calibration helper
- iterative model calibration in addition to global calibration
- generation of calibration baskets for bermudan non standard swaptions