Design Patterns for Algorithmic Differentiation

Dr. Sebastian Schlenkrich
QuantLib Workshop, November 2013

Joint work with Jan Riehme, RWTH Aachen
Agenda

1. Sensitivities of Exotic Derivatives

2. Algorithmic Differentiation (AD) at a Glance

3. Incorporation of AD Methodologies into Financial Libraries

4. Proof of Concept for Bermudan Swaption Vega in QuantLib
Sensitivities of Exotic Derivatives
Generic Valuation Process for Exotics

1. Market Quotes
2. Model Calibration
3. Model Parameters
4. Numerical Method
5. Exotics Valuation

Example: Bermudan Swaptions with Hull White Model

1. Swap-Vol
2. Market vs. HW Price
3. Short Rate Volatility
4. Backward Lattice
5. Bermudan Price
Notations and Mappings

Example: Bermudan Swaptions with Hull White Model

\[ \sigma_{B76} \quad \text{Market}(\sigma_{B76}) \quad \sigma_{HW} \quad \text{Exotic}(\sigma_{HW}) \quad V \]

- Vector of Swaption volatilities \( \sigma_{B76} \)
- Vector of Hull White short rate volatility term structure \( \sigma_{HW} \)
- Vector functions for Swaption prices \( \text{Market}(\sigma_{B76}) \) and \( \text{Model}(\sigma_{HW}) \) using Black’76 and Hull White analytical model formulas
- Exotics valuation function \( V = \text{Exotic}(\sigma_{HW}) \) based on model parameters
Exotics Sensitivity Evaluation

- Assume invertability and differentiability of functions involved

\[
\begin{align*}
\sigma_{B76} & \xrightarrow{\text{Market}(\cdot)} \text{Swpt} & \xrightarrow{\text{Model}^{-1}(\cdot)} \sigma_{HW} & \xrightarrow{\text{Exotic}(\cdot)} V \\
\frac{dV}{d\sigma_{B76}} &= \text{Exotic}'(\sigma_{HW}) \cdot \text{Model}'(\sigma_{HW})^{-1} \cdot \text{Market}'(\sigma_{B76})
\end{align*}
\]

\[
V = \text{Exotics} \left( \text{Model}^{-1}(\text{Market}(\sigma_{B76})) \right)
\]
Sensitivities Involved

\[ \frac{dV}{d\sigma_{B76}} = Exotic'(\sigma_{HW}) \cdot Model'(\sigma_{HW})^{-1} \cdot Market'(\sigma_{B76}) \]

- Analytic Vega formula
- Analytic expression available, but tedious
- More complex models than Hull White may not allow analytic derivative formulas

- Numerical method in general does not exhibit analytic derivative
- Remedies:  
  - Finite difference approximations
  - Algorithmic differentiation
Algorithmic Differentiation at a Glance
Algorithmic Differentiation (AD)

- Principles and techniques to augment computer models
- Sensitivities of output variables with respect to inputs of the model
- Numerical values rather than symbolic expressions
- Sensitivities exact up to machine precision (no rounding/cancellation errors)
- Apply chain rule of differentiation to operations like „+“, „*“, „exp ( )“, ...
Example: Black’76 Vega of ATM Option

\[ V = F \left[ 2N \left( \sigma \sqrt{T}/2 \right) - 1 \right], \quad \frac{dV}{d\sigma} = F \phi \left( \sigma \sqrt{T}/2 \right) \sqrt{T} \]

**Single Assignment Code of Elementary Operations**

<table>
<thead>
<tr>
<th>Initialisation</th>
<th>( F, \sigma ) and ( T )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluation</td>
<td>( v_1 = \sqrt{T} )</td>
</tr>
<tr>
<td></td>
<td>( v_2 = \sigma \cdot v_1 )</td>
</tr>
<tr>
<td></td>
<td>( v_3 = v_2/2 )</td>
</tr>
<tr>
<td></td>
<td>( v_4 = N(v_3) )</td>
</tr>
<tr>
<td></td>
<td>( v_5 = 2 \cdot v_4 )</td>
</tr>
<tr>
<td></td>
<td>( v_6 = v_5 - 1 )</td>
</tr>
<tr>
<td></td>
<td>( v_7 = F \cdot v_6 )</td>
</tr>
</tbody>
</table>

**Original Computer Model**

| \( \dot{F} = 0, \dot{\sigma} = 1 \) and \( \dot{T} = 0 \) |
| \( \dot{v}_1 = 1/(2v_1) \)                                 |
| \( \dot{v}_2 = \dot{\sigma} \cdot v_1 + \sigma \cdot \dot{v}_1 \) |
| \( \dot{v}_3 = \dot{v}_2/2 \)                             |
| \( \dot{v}_4 = \phi(v_3) \cdot \dot{v}_3 \)              |
| \( \dot{v}_5 = 2 \cdot \dot{v}_4 \)                      |
| \( \dot{v}_6 = \dot{v}_5 \)                               |
| \( \dot{v}_7 = \dot{F} \cdot v_6 + F \cdot \dot{v}_6 \)  |

| Augmented Computer Model |
| \( \dot{V} = \dot{v}_7 \) |
Implementation and Tools

Methodologies

Source Code Transformation
- Applied to the model code in compiler fashion
- Generate AD model as new source code
- Original code may need to be adapted slightly to meet capabilities of AD tool

Operator Overloading
- provide new (active) data type
- Overload all relevant operators/functions with sensitivity aware arithmetic
- AD model derived by changing intrinsic to active data type

Some Tools for C++
- ADIC2, dcc, TAPENADE
- ADOL-C, dco, ADMB/AUTODIF
Some References for Automatic Differentiation

Community Website

www.autodiff.org

Standard Text Book


Recent Practitioner’s Text Book

Incorporation of AD into Financial Libraries
Practical Considerations for AD in Software Packages

- Source transformation and overloading result in new AD-enabled model
- AD-enabled model needs to be maintained consistently in software development cycle besides original model
  - E.g. by re-creation of AD model after each original model update
- AD model usually does not implement the interface of the original model
  - Sensitivity evaluation needs to be wrapped appropriately
Example: Bermudan Swaption with Hull White Model

Hull White Model Valuation

• European Swaptions as European Coupon Bond Options (CBO)
• Bermudan Swaptions as Bermudan CBO

Hull White Model Vegas

• \(d \left[ \text{Europ. CBO Price} \right] / d \left[ \text{short rate volatility} \right] = Model'(\sigma_{HW})\)
• \(d \left[ \text{Berm. CBO Price} \right] / d \left[ \text{short rate volatility} \right] = Exotic'(\sigma_{HW})\)

Operator Overloading AD Tool ADTAGEO*

• Algorithmic Differentiation Through Automatic Graph Elimination Ordering
• Sensitivity aware user defined data type daglad
• Sensitivity \(dy/dx\) for \(y = f(x)\) via \(\%\) operator, that is \(dydx = y \% x\)

template<class \texttt{DataType}, class \texttt{PassiveType}, class \texttt{ActiveType}>
oclass TemplateHullWhiteModel 
{
    std::vector<\texttt{DataType}> volaDates;
    std::vector<\texttt{ActiveType}> volaValues;
    \texttt{PassiveType} meanReversion;
    ...
    virtual \texttt{ActiveType} CouponBondOption(...);
    virtual \texttt{ActiveType} BermudanBondOption(...);
};
Object Adapter Design Pattern

```
<<bind>>
< DateType->double, PassiveType->double, ActiveType->double >

Double HullWhiteModel

Adtageo HullWhiteModel

CBOVegas(...) BermudanVegas(...)

Daglad HullWhiteModel

<<bind>>
< DateType->double, PassiveType->double, ActiveType->daglad >

Template HullWhiteModel

CouponBondOption(...) BermudanBondOption(...)
```
class AdtageoHullWhiteModel : public DoubleHullWhiteModel {
    DagladHullWhiteModel *aModel;
    std::vector<double> bermudanVegas;
    ...
    virtual double BermudanBondOption(...) {
        daglad res = aModel->BermudanBondOption(...);
        for (size_t i=0; i<bermudanVegas.size(); ++i)
            bermudanVegas[i] = res % aModel->volaValues[i];
        return res.val();
    }
    virtual std::vector<double> BermudanVegas() {
        return bermudanVegas;
    }
};
Flexible Incorporation into QuantLib Framework

- Map instrument to Bermudan CBO
- Calibrate Hull White Model
- Evaluate NPV
- Try downcast and request Vegas
Generalization for Several AD Tool Implementations

TemplateModel

DoubleModel

<< bind >>
< ActiveType->double >

ADModel

<< bind >>
< ActiveType->ADType1 >

ADImpl1Model

adopts

1

ADImpl2Model

adopts

1

<< bind >>
< ActiveType->ADType2 >

ADImpl1Model

ActiveType
Proof of Concept for Bermudan Swaption Vega in QuantLib
# QuantLib Object Setup in Excel

## HullWhiteModel

<table>
<thead>
<tr>
<th>Error</th>
<th>HullWhiteModel#0007</th>
</tr>
</thead>
<tbody>
<tr>
<td>ObjectID</td>
<td>HullWhiteModel</td>
</tr>
<tr>
<td>DiscountCurve</td>
<td>6M-EUR-Swap-Curve#0000</td>
</tr>
<tr>
<td>MeanReversion</td>
<td>0.084</td>
</tr>
<tr>
<td>VolaTimes</td>
<td>0</td>
</tr>
<tr>
<td>VolaValues</td>
<td>0.01</td>
</tr>
<tr>
<td>Permanent</td>
<td></td>
</tr>
<tr>
<td>Trigger</td>
<td></td>
</tr>
<tr>
<td>OverWrite</td>
<td></td>
</tr>
</tbody>
</table>

## AD HullWhiteModel

<table>
<thead>
<tr>
<th>Error</th>
<th>ADHullWhiteModel#0007</th>
</tr>
</thead>
<tbody>
<tr>
<td>ObjectID</td>
<td>ADHullWhiteModel</td>
</tr>
<tr>
<td>DiscountCurve</td>
<td>6M-EUR-Swap-Curve#0000</td>
</tr>
<tr>
<td>MeanReversion</td>
<td>0.084</td>
</tr>
<tr>
<td>VolaTimes</td>
<td>0</td>
</tr>
<tr>
<td>VolaValues</td>
<td>0.008749649</td>
</tr>
<tr>
<td>Permanent</td>
<td></td>
</tr>
<tr>
<td>Trigger</td>
<td></td>
</tr>
<tr>
<td>OverWrite</td>
<td></td>
</tr>
</tbody>
</table>

## BondOptionEngine

<table>
<thead>
<tr>
<th>Error</th>
<th>BondOptionEngineSwaption#0008</th>
</tr>
</thead>
<tbody>
<tr>
<td>ObjectID</td>
<td>BondOptionEngineSwaption</td>
</tr>
<tr>
<td>HullWhiteModel</td>
<td>ADHullWhiteModel#0007</td>
</tr>
<tr>
<td>Dimension</td>
<td>1001</td>
</tr>
<tr>
<td>GridRadius</td>
<td>0.3</td>
</tr>
<tr>
<td>BermudanTolerance</td>
<td>1.00E-04</td>
</tr>
<tr>
<td>SwaptionProperties</td>
<td>SwaptionProperties#0007</td>
</tr>
<tr>
<td>CalibrationTolerance</td>
<td>1.00E-10</td>
</tr>
<tr>
<td>Permanent</td>
<td></td>
</tr>
<tr>
<td>Trigger</td>
<td></td>
</tr>
<tr>
<td>OverWrite</td>
<td></td>
</tr>
</tbody>
</table>

## SetPricingEngine

<table>
<thead>
<tr>
<th>SetPricingEngine</th>
<th>TRUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>NPV</td>
<td>0.028766898</td>
</tr>
<tr>
<td>ErrorEstimate</td>
<td>6.54674E-06</td>
</tr>
<tr>
<td>Bermudan Vega</td>
<td>19.20%</td>
</tr>
<tr>
<td>EstimateAccuracy</td>
<td>TRUE</td>
</tr>
<tr>
<td></td>
<td>TRUE</td>
</tr>
</tbody>
</table>
### Detailed QuantLib Valuation Results in Excel

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>30.11.2011</td>
<td>27.79%</td>
<td>1.727%</td>
<td>9.77%</td>
<td>1.0</td>
<td>1.311%</td>
<td>1.727%</td>
<td>1.727%</td>
<td>3.29%</td>
</tr>
<tr>
<td>30.11.2012</td>
<td>26.13%</td>
<td>1.885%</td>
<td>12.16%</td>
<td>2.0</td>
<td>1.233%</td>
<td>1.885%</td>
<td>1.885%</td>
<td>3.49%</td>
</tr>
<tr>
<td>29.11.2013</td>
<td>24.27%</td>
<td>1.724%</td>
<td>12.84%</td>
<td>3.0</td>
<td>1.103%</td>
<td>1.724%</td>
<td>1.724%</td>
<td>2.69%</td>
</tr>
<tr>
<td>28.11.2014</td>
<td>22.63%</td>
<td>1.507%</td>
<td>12.60%</td>
<td>4.0</td>
<td>0.997%</td>
<td>1.507%</td>
<td>1.507%</td>
<td>2.04%</td>
</tr>
<tr>
<td>30.11.2015</td>
<td>21.42%</td>
<td>1.309%</td>
<td>11.80%</td>
<td>5.0</td>
<td>0.950%</td>
<td>1.309%</td>
<td>1.309%</td>
<td>1.67%</td>
</tr>
<tr>
<td>30.11.2016</td>
<td>20.83%</td>
<td>1.150%</td>
<td>10.62%</td>
<td>6.0</td>
<td>0.994%</td>
<td>1.150%</td>
<td>1.150%</td>
<td>1.64%</td>
</tr>
<tr>
<td>30.11.2017</td>
<td>20.12%</td>
<td>0.944%</td>
<td>9.02%</td>
<td>7.0</td>
<td>0.905%</td>
<td>0.944%</td>
<td>0.944%</td>
<td>1.36%</td>
</tr>
<tr>
<td>30.11.2018</td>
<td>19.75%</td>
<td>0.743%</td>
<td>7.14%</td>
<td>8.0</td>
<td>0.930%</td>
<td>0.743%</td>
<td>0.743%</td>
<td>1.27%</td>
</tr>
<tr>
<td>29.11.2019</td>
<td>19.16%</td>
<td>0.504%</td>
<td>4.97%</td>
<td>9.0</td>
<td>0.821%</td>
<td>0.504%</td>
<td>0.504%</td>
<td>0.97%</td>
</tr>
<tr>
<td>30.11.2020</td>
<td>18.96%</td>
<td>0.267%</td>
<td>2.59%</td>
<td>10.0</td>
<td>0.876%</td>
<td>0.267%</td>
<td>0.267%</td>
<td>0.78%</td>
</tr>
<tr>
<td><strong>Sum</strong></td>
<td><strong>19.20%</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bermudan NPV: 2.877%

Vega as 1 unit shift sensitivity

A flat 1% shift in Swaption volatilities yields a 0.192% Bermudan NPV shift
Conclusions
Conclusions

- Market sensitivities for Exotics can be evaluated by differentiating calibration and Exotics instrument model pricers
- Algorithmic Differentiation (AD) methodologies yield accurate sensitivities
- Model templatisation and object adapter design patterns are flexible concepts to incorporate Operator Overloading AD methodologies

QuantLib should make use of template-based model implementations
Your Contacts at d-fine

Dr Sebastian Schlenkrich
Manager
Mobile +49 162 263-1525
E-Mail: Sebastian.Schlenkrich@d-fine.de

Dr Mark Beinker
Head of Business Unit Applied Financial Engineering
Partner
Tel +49 69 90737-305
E-Mail: Mark.Beinker@d-fine.de

d-fine GmbH
Opernplatz 2
60313 Frankfurt am Main
Germany
Tel +49 69 90737-0
www.d-fine.de

d-fine Ltd
6-7 Queen St
London
EC4N 1SP
Tel +44 20 7776-1000
www.d-fine.co.uk