Large Project 2009

This year's Large Project will be a joint project for all students participating in the course. In this project we shall write a "professional" package for evaluating contingent claims on the fixed income market. The term "professional" means that, contrary to the standard courseworks, this software has to be written for an end-user who uses financial instruments in its everyday operations and is not an expert in numerical analysis and computational finance. This means that we have to feed the system with data which are readily available from Bloomberg or Reuters services. Finally, we have to offer instruments and methods which are in demand, i.e. industry standards and not "white elephants".

We shall write the project as a library of functions performing different tasks. We have also to write a main module (main programme) which manages all functions, supplies data to them, provides the communication between the functions and solves the problem of communication with a future end-user interface.

To make the project accessible we will limit our interest to most popular simple instruments: swaps, bond options, caps, floors and swaptions. In evaluating derivative instruments we shall explore the Black formula for caplets and swaptions and if necessary use simple short rate models and lattice numerical methods.

In the course of the project the following tasks are essential:

1. Construction of the main skeleton of the software package. The package will consist of a number of modules. Which modules to construct and how to solve the problem of communication between these modules are absolutely essential.

2. Construction of the calendar. Financial transactions are settled for working days. Working days are specific for every market, i.e. currency of denomination and geographical location – PLN denominated instruments are traded in Warsaw, but also in London, and USD denominated instruments are traded in New York, London, Tokyo etc. and all these are different markets. To calculate the price of an instrument which contains many cash flows in different time moments we have to know exact moments of these cash flows. Usually instruments are defined by some general rules, say the coupon days for a bond coupons come every 6 months from the date of issue. Knowing the issue date of a bond, we can calculate the dates for coupon dates. In some markets these days have to be working days. When a certain date from this list is not a working day we have to use a rule specific for a given market saying which working day to use as the coupon date. But then come the value date, the date in which the actual cash flow takes place. This has to be a working day (in all markets) and again a specific rule says how to calculate the value date knowing the coupon date. Implementing all these rules even for a single instrument on a given market is quite complicated. We have to implement the following functions (to simplify the project for a single market only – the Polish market is a first choice):

   (a) Construction of a standard calendar function (or functions). This function has to perform following operations: i) given a fixed date, calculate which day of the week is this date, ii) how many days are from a fixed date to any other specific date, iii) given a fixed date supply the date distant by n days.

   (b) Construction of a "working days" function. This function has to be specific for a given market. It has to recognize holidays: Saturdays and Sundays, but also state holidays (specific for a country) and days which are financial market non-working days ("Bank holidays"), say Christmas Eve, Good Friday (in countries in which it is not a state holiday) or May, 2 in Poland, i.e. days in which a local stock exchange is not working.

   (c) Implementation of the day-count-convention – the rule which says how many days a year has and how a month is defined. This rule is specific for a market and for an instrument, i.e. on the same market two instruments can be constructed with different day-count-conventions.
   (d) Implementation of the rule which defines how the contract date (for example coupon date) is modified if it is not a working day (if applicable).
   (e) Implementation of the rule defining how many days from the contract date is the value date and how the value date is modified if it is not a working day.

3. Processing input data – term structure of interest rates. Recovering the term structure of interest rates from the prices of traded instruments (bonds or swaps) is complicated because it requires full implementation of the calendar functions. We shall base on a term structure of discount factors where discount factors are quoted together with dates to which they refer. Such a term structure solves for us the calendar problem: we can calculate spot and forward interest rates exactly for all dates. We need only proper interpolation of rates which can be tricky (see the paper by W. Walus). Besides this type of data is offered by the Reuters service so our software will be based on available market data.

4. Processing input data – term structure of volatilities. Available volatilities are implied cap (or swaption) volatilities. We need caplet volatilities for different time moments and periods. Recovering caplet volatilities from cap volatilities is a complicated operation (there are several models which have to be implemented). The good reference is Section 6.4 and 6.5 of the book by Brigo and Mercurio. Recovering volatility structure from swaption quotations is even more complicated. We shall stick to a simple approach very similar to that used for caps (see Section 6.7 of the book by Brigo and Mercurio).

5. Pricing swaps, options on zero coupon and coupon bonds, caps, floors and swaptions by the Black formula using obtained term structure of interest rates and volatilities. This pricing module should include possibilities for pricing also more sophisticated instruments like callable/putable swaps, variable notional principal swaps, callable bonds, etc.

6. Calibration of interest-rate models to data. The following interest-rate models will be considered (which model we implement in practice is not very critical, at least models a), b) and c) should be implemented; on the other hand it would be good to construct a module in which adding some new models will be easy):

   (a) Hull-White model (Vasicek model and shift-extended Vasicek model are special cases);
   (b) Black-Karasinski model;
   (c) Black-Derman-Toy model as a special case of Black-Karasinski model;
   (d) CIR model;
   (e) shift-extended CIR model (so called CIR++ model).

7. Construction of a lattice. Calibration is in fact trough the construction of the lattice, but same lattice can be suitable for several models. The following lattices will be considered (again which lattice we implement in practice is not very critical, at least lattice a) and b) should be implemented):

   (a) Hull-White trinomial lattice;

(b) Black-Derman-Toy binomial lattice;

(c) Schmidt lattice.

8. Pricing on a lattice options on zero coupon and coupon bonds, caps, floors and swaptions.

9. Documentation. It is absolutely essential to describe carefully the theoretical background of every function, say which are the calendar rules in the Polish market, how to calibrate the Hull-White model to the term structures of interest rates and volatilities, how to get the caplets volatilities from the given cap volatilities, etc. But also the documentation of the computer code has to be provided. A particular attention should be put on the structure of the programme (block scheme), the format of input and output data, signaling choices, etc.

The work on the project should start with a general information on short interest rate models (see Lecture 10, a good reference is also Chapter 1 and 3 of the book by Brigo and Mercurio).

The project will consist of the following tasks:

1. Main programme.
2. Calendar functions.
3. Processing input data.
4. Pricing instruments in Black's model (Black formula).
5. Construction and calibration of lattices.
6. Pricing instruments on a lattice.

Additional literature comments

- Calendar functions — main source of information are web pages, but you can also consult the course coordinators.
- Pricing instruments in Black's model — general idea is in the book by Brigo and Mercurio, specifications of instruments which should be included in the pricing module will be provided by the course coordinators.
- Construction of a lattice, calibration of a given model on the constructed lattice and pricing of derivative instruments:
  - Hull-White trinomial lattice applied to Hull-White, Black-Karasinski and CIR models — main reference is the paper by Leippold and Wiener, but original papers by Hull and White can be helpful.
  - Black-Derman-Toy binomial lattice — main reference is Chapter 11 (Sections 11.1–11.3) of the book by London, the original paper by Black, Derman and Toy can also be a good reference.
  - Schmidt lattice — the reference is the original paper by Schmidt.

– The implementation of shift-extended Vasicek and CIR models is not discussed explicitly in papers devoted to the lattice construction. From the technical point of view these models are embedded in other models. But they require special treatment in calibration. The models are thoroughly discussed in the book by Brigo and Mercurio. Construction and calibration of corresponding lattices is well described in the book by London (Sections 11.9–11.11).

References

1. Damian Brigo and Fabio Mercurio – *Interest Rate models – Theory and Practice*, Springer 2006. (Library)

2. Fischer Black, Emanuel Derman and William Toy – A one-factor model of interest rates and its application to treasury bond options, *Financial Analysts Journal*, **46** no 1 (Jan-Feb 1990), 33–39. (pdf)

3. Markus Leippold and Zvi Wiener – Efficient calibration of trinomial trees for one-factor short rate models, *Review of Derivatives Research*, **7** (2004), 213–239. (pdf)

4. Justin London – *Modeling Derivatives in C++*, Wiley 2005. (Coordinators)

5. W.M. Schmidt – On a general class of one-factor models for the term structure of interest rates, *Finance and Stochastics*, **1** (1997), 3–24. (pdf)

6. Wlodzimierz Walus – On approximation of discount factors and yields. (pdf)