Source | Option Family

pad

1 Description

Premia 5

This family is split in the LookBack family and the Asian family.

1.1 LookBack Options

The payoff is given by:
$\varphi\left(S_T, M_{T_0,T_1}\right), or\ \left(S_T, m_{T_0,T_1}\right)$ which is paid at the maturity T where M is the maximum and m the minimum of the underlying over the period $[T_0, T_1]$. Two varieties exist:

- Total LookBack
  Then M or m are computed over the whole life of the option, ie T_0 and T_1 are irrelevant.

- Partial LookBack
  M or m are computed betwen the starting date T_0 and the final date T_1 with $T_1 \leq T$. Note that T_0 may be less than the pricing date or greater than the pricing date (forward-starting case). A partial LookBack may be Continuous or Discrete, in which case the maximum M or the minimum m is taken from a regular sampling of the underlying between T_0 and T_1, which is specified by the number SamplingDates.

1.2 Asian Options

The payoff is given by: $\varphi\left(S_T, A_{T_0,T_1}\right)$, which is paid at the maturity T where A is the average of the underlying over the period $[T_0, T_1]$. Two varieties exist:

- Forward-Starting Asian
  T_0 is greater or equal to the pricing date.
- Backward-Starting Asian
  T_0 is less than or equal to the pricing date.

An asian option may be Continuous or Discrete, in which case the average A is taken from a regular sampling of the underlying between T_0 and T_1, which is specified by the number SamplingDates.

2 Code Implementation

```
#ifndef  _PAD_H
#define _PAD_H

#include  "optype.h"
#include  "var.h"

#include  "chk.h"
#include  "numfunc.h"

#define TYPEOPT PAD

/*PathDep Option*/
typedef struct TYPEOPT{
VAR      PayOff; /* The Payoff is phi(stock,path_dep) */
VAR   MinOrElse; /* cf supra*/
VAR   EuOrAm;
VAR   PartOrTot; /* Partial or total pathdep:

a partial pathdep is specified
by starting_date, final_date*/

VAR   ContOrDisc; /*Continuous or Discrete:
a discrete pathdep is specified
by frequency (regular sampling) */

VAR  PathDep; /* The PathDep functional definition:

new_path-dep=psi(PathDep->Par,stock,time)
```

```
where:

starting_date is in PathDep->Par[0],
final_date is in PathDep->Par[1],
frequency  is in PathDep->Par[2],
initial_path_dep is in PathDep->Par[3],
current_path_dep is in PathDep->Par[4]

  !!!!!WARNING!!!!!
 Wether the pathdep is backard/forward
 should be tested in ChkOpt
*/
VAR                                  Maturity;
} TYPEOPT;


/*MinOrElse*/
#define MINIMUM 0
#define MAXIMUM 1
#define AVERAGE 2

int OPT(Get)(int user,Planning *pt_plan,Option *opt);
int OPT(Show)(int user,Planning *pt_plan,Option *opt);
int OPT(Check)(int user,Planning *pt_plan,Option *opt);

#endif
```