

# doublim

## 1 The DoubleLimit Family

### Premia 22

For this family, the payoff is given by:

- For an Out option:

$\varphi(S_T)$  if  $S$  doesn't reach the upper boundary  $U(t)$  or the lower boundary  $L(t)$  between the pricing date  $t_0$  and the maturity  $T$ .

$R(t)$ , which is paid at the time  $t$  when the upper or lower barrier is reached. The rebate  $R(t)$  is the same for the two barriers.

The payoff  $H_T$  in monetary unit of time of the maturity  $T$  may be written:

$$H_T = \varphi(S_T) \mathbf{1}(\tau^* > T) + e^{r(T-\tau^*)} R(\tau^*) \mathbf{1}(\tau^* \leq T)$$

under the assumption the instantaneous interest rate  $r$  is constant, where

$$\tau^* = \inf \{u > t_0, S_u > U(u) \text{ or } S_u < L(u)\}$$

It is assumed that  $L(t_0) < S_{t_0} < U(t_0)$ .

- For an In option:

The payoff  $H_T$  in monetary unit of time of the maturity  $T$  may be written:

$$H_T = \varphi(S_T) \mathbf{1}(\tau^* \leq T) + R(T) \mathbf{1}(\tau^* > T)$$

## 2 Code Implementation

```
#ifndef _DOUBLIM_H
#define _DOUBLIM_H

#include "optype.h"
#include "var.h"
#include "chk.h"
#include "numfunc.h"
#include "option.h"

#define TYPEOPT DOUBLIM

/*Limit Option// General double barrier*/

typedef struct TYPEOPT
{
    /* var_setable */
    VAR    PayOff;
    VAR    Rebate;
    VAR    LowerLimit;
    VAR    UpperLimit;
    VAR    Maturity;
    VAR    DateBetween0andMaturity ;
    /* var_fixed */
    VAR    OutOrIn;
    VAR    Parisian;
    VAR    TwoDoubleStep;
    VAR    RebOrNo;
    VAR    EuOrAm;

} TYPEOPT;

int OPT(Get)(int user, Planning *pt_plan, Option *opt, Model *mod);
int OPT(FGet)(char **InputFile, int user, Planning *pt_plan, Option *opt, Model
int OPT(Show)(int user, Planning *pt_plan, Option *opt, Model *mod);
int OPT(Check)(int user, Planning *pt_plan, Option *opt);

#endif
```