

Help

```
/*-----*/
/* Monte Carlo algorithm for caplet prices in one-factor LMM with jumps */
/* Algorithm of Glasserman/Merener */
/* */
/*-----*/
/* Sonke Blunck, Premia 2005 */
/*-----*/

#include "
href../../mod/lmm_jump1d/lmm_jump1d_std/glassermanmerener_h_src.pdfglasserma

extern "C" {
#include "
href../../mod/lmm_jump1d/lmm_jump1d_std/lmm_jump1d_std_h_src.pdf
#include "
href../../common/enums_h_src.pdfenums.h"

#if defined(PremiaCurrentVersion) && PremiaCurrentVersion < (2007+2) //The "#els
static int CHK_OPT(MC_GM)(void *Opt, void *Mod)
{
return NONACTIVE;
}
int CALC(MC_GM)(void *Opt, void *Mod, PricingMethod *Met)
{
return AVAILABLE_IN_FULL_PREMIA;
}
#else

static int mc_glassermanmerenenr_caplet(NumFunc_1 *p, double l0, double t0, do
{

capletMat = capletMat - t0;

return lmm_jump_caplet_MC_pricer(tenor, capletMat, strike, l0, sigma, number
}

int CALC(MC_GM)(void *Opt, void *Mod, PricingMethod *Met)
{
TYPEOPT *ptOpt = (TYPEOPT *)Opt;
```

```

    TYPEMOD *ptMod = (TYPEMOD *)Mod;
    int init_mc;
    init_mc = pnl_rand_init(Met->Par[0].Val.V_ENUM.value, 1, Met->Par[1].Val.V_L
    if (init_mc != OK) return init_mc;
    return mc_glassermanmerenenr_caplet(ptOpt->PayOff.Val.V_NUMFUNC_1, ptMod->10
                                     ptMod->T.Val.V_DATE,
                                     ptMod->Sigma.Val.V_PDOUBLE,
                                     ptOpt->BMaturity.Val.V_DATE,
                                     ptOpt->FixedRate.Val.V_PDOUBLE,
                                     ptOpt->ResetPeriod.Val.V_DATE,
                                     Met->Par[0].Val.V_ENUM.value,
                                     Met->Par[1].Val.V_LONG,
                                     &(Met->Res[0].Val.V_DOUBLE));

}

static int CHK_OPT(MC_GM)(void *Opt, void *Mod)
{

    if ((strcmp(((Option *)Opt)->Name, "Caplet") == 0))
        return OK;
    else
        return WRONG;
}

#endif //PremiaCurrentVersion
static int MET(Init)(PricingMethod *Met, Option *Opt)
{
    if (Met->init == 0)
    {
        Met->init = 1;

        Met->Par[0].Val.V_ENUM.value = 0;
        Met->Par[0].Val.V_ENUM.members = &PremiaEnumRNGs;
        Met->Par[1].Val.V_LONG = 100;
    }
    return OK;
}

PricingMethod MET(MC_GM) =
{

```

```

"MC_GlassermanMerener",
{ {"RandomGenerator", ENUM, {100}, ALLOW},
  {"N Simulation", LONG, {100}, ALLOW},
  {" ", PREMIA_NULLTYPE, {0}, FORBID}
},
CALC(MC_GM),
{{"Price", DOUBLE, {100}, FORBID}/*,{"Delta",DOUBLE,{100},FORBID}*/ , {" "},
CHK_OPT(MC_GM),
CHK_ok,
MET(Init)
} ;
}

```