

## [Help](#)

```
extern "C" {
#include "
href../../mod/temperedstable1d/temperedstable1d_vol/temperedstable1d_vol_h_sr
}
#include "
href../../common/math/numerics_h_src.pdfmath/numerics.h"
extern "C" {

#if defined(PremiaCurrentVersion) && PremiaCurrentVersion < (2008+2) //The "#els
static int CHK_OPT(CF_CGMY_VARIANCESWAP)(void *Opt, void *Mod)
{
    return NONACTIVE;
}
int CALC(CF_CGMY_VARIANCESWAP)(void *Opt, void *Mod, PricingMethod *Met)
{
    return AVAILABLE_IN_FULL_PREMIA;
}
#else
//-----
static int ap_cgmy_varswap_cf(double S0, double Strike, double T, double r, do
{
    double K;

    double gamma2p, gamma2m;

    K = Strike;

    gamma2p = pnl_tgamma(2.0 - ap);
    gamma2m = pnl_tgamma(2.0 - am);
    double lpnu = exp((2.0 - ap) * log(lap));
    double lmnu = exp((2.0 - am) * log(lam));

    double mval = cpp * gamma2p / lpnu + cmm * gamma2m / lmnu;

    *fairval = sqrt(mval) * 100.0;
    *ptprice = exp(-r * T) * (mval * 10000 - K * K);

    return OK;
}
```

```

int CALC(CF_CGMY_VARIANCESWAP)(void *Opt, void *Mod, PricingMethod *Met)
{
    TYPEOPT *ptOpt = (TYPEOPT *)Opt;
    TYPEMOD *ptMod = (TYPEMOD *)Mod;
    double r, divid, strike, spot;
    NumFunc_1 *p;

    r = log(1. + ptMod->R.Val.V_DOUBLE / 100.);
    divid = log(1. + ptMod->Divid.Val.V_DOUBLE / 100.);
    p = ptOpt->PayOff.Val.V_NUMFUNC_1;
    strike = p->Par[0].Val.V_DOUBLE;
    spot = ptMod->S0.Val.V_DOUBLE;

    return ap_cgmy_varswap_cf(
        spot, strike, ptOpt->Maturity.Val.V_DATE - ptMod->T.Val.V_DATE, r,
        &(Met->Res[0].Val.V_DOUBLE), &(Met->Res[1].Val.V_DOUBLE));
}

static int CHK_OPT(CF_CGMY_VARIANCESWAP)(void *Opt, void *Mod)
{
    if ((strcmp(((Option *)Opt)->Name, "VarianceSwap") == 0))
        return OK;

    return WRONG;
}

#endif //PremiaCurrentVersion
static int MET(Init)(PricingMethod *Met, Option *Opt)
{
    static int first = 1;

    if (first)
    {
        first = 0;
    }
    return OK;
}

PricingMethod MET(CF_CGMY_VARIANCESWAP) =

```

```

{
    "CF_CGMV_VARIANCESWAP",
    { {" " , PREMIA_NULLTYPE, {0}, FORBID}},
    CALC(CF_CGMV_VARIANCESWAP),
    { {"Fair strike in annual volatility points", DOUBLE, {100}, FORBID},
      {"Price in 10000 variance points", DOUBLE, {100}, FORBID},
      {" " , PREMIA_NULLTYPE, {0}, FORBID}
    },
    CHK_OPT(CF_CGMV_VARIANCESWAP),
    CHK_ok ,
    MET(Init)
} ;

/*////////////////////////////////////////*/
}

```