

[Help](#)

```

#include "hawkes_trading_stdtd.h"
#include "stdtd/stdtd.h"

#if defined(PremiaCurrentVersion) && PremiaCurrentVersion < (2015+2) //The "#els
static int CHK_OPT(CF_AlfonsiBlanc)(void *Opt, void *Mod)
{
    return NONACTIVE;
}

int CALC(CF_AlfonsiBlanc)(void *Opt, void *Mod, PricingMethod *Met)
{
    return AVAILABLE_IN_FULL_PREMIA;
}
#else

/*////////////////////////////////////*/
double Theoretical_Cost(double T,double X0,double rho,double nu,double epsilon,d

double m2,cost,delta,Sigma,G0,resfact,epsfact,mg_dev,dr_ratio,eT,gT;
double rho2,rho3,rho4,rho5,T2,T3,T4,T5;

if(MAX(ioc,fabs(beta-ios))>0.00000001) return 0.0;

m2 = 2*m1*m1;
delta = kp0-km0;
Sigma = kp0+km0;
G0 = 1.0+0.5*nu*rho*T;
resfact = 1.0/(2.0+rho*T);
epsfact = 1.0/(1.0-epsilon);
dr_ratio = delta*m1/rho;
mg_dev = q*D0-G0*dr_ratio;

rho2 = rho*rho;
rho3 = rho2*rho;
rho4 = rho3*rho;
rho5 = rho4*rho;
T2 = T*T;
T3 = T2*T;
T4 = T3*T;

```

```

T5 = T4*T;

cost = ((1.0-epsilon)*resfact+0.5*epsilon)*X0*X0;
cost += rho*T*resfact*mg_dev*X0;
cost -= epsfact*0.5*rho*T*resfact*mg_dev*mg_dev;
cost -= epsfact*nu*nu*rho3*T3*dr_ratio*dr_ratio/48.0;

eT = -(1.0-nu)*(1.0-nu)*epsfact*(m2-beta*m1*m1*(2*rho-beta)/rho2)*(0.5*T-log(1.0
eT += beta*nu*(1.0-nu)*m1*m1*epsfact/(4.0*rho2)*(1.0-beta/rho)*rho2*T2;
eT -= beta*beta*nu*nu*m1*m1*epsfact/(8.0*rho3)*(rho2*T2+rho3*T3/3.0+rho4*T4/24.0
cost += eT*Sigma;

gT = -2.0*beta*k_infty*(1.0-nu)*(1.0-nu)*epsfact*(m2-beta*m1*m1*(2*rho-beta)/rho
gT += beta*beta*k_infty*nu*(1.0-nu)*m1*m1*epsfact/(6.0*rho3)*(1.0-beta/rho)*rho3
gT -= beta*beta*beta*k_infty*nu*nu*m1*m1*epsfact/(12.0*rho4)*(rho3*T3+rho4*T4/4.
cost += gT;

cost /= q;
cost -= (S0+D0)*X0;

*ptprice=cost;

return OK;
}

int CALC(CF_AlfonsiBlanc)(void *Opt, void *Mod, PricingMethod *Met)
{
    TYPEOPT *ptOpt = (TYPEOPT *)Opt;
    TYPEMOD *ptMod = (TYPEMOD *)Mod;

    return Theoretical_Cost(ptOpt->Maturity.Val.V_DATE,ptOpt->InitialPosition.Val.
}

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

    return WRONG;
}

```

```
#endif //PremiaCurrentVersion

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

    return OK;
}

PricingMethod MET(CF_AlfonsiBlanc) =
{
    "CF_AlfonsiBlanc",
    {{" ", PREMIA_NULLTYPE, {0}, FORBID}},
    CALC(CF_AlfonsiBlanc),
    { {"Cost", DOUBLE, {100}, FORBID},
      {" ", PREMIA_NULLTYPE, {0}, FORBID}
    },
    CHK_OPT(CF_AlfonsiBlanc),
    CHK_ok,
    MET(Init)
};
```