

# NAG Library Routine Document

## S30NCF

**Note:** before using this routine, please read the Users' Note for your implementation to check the interpretation of *bold italicised* terms and other implementation-dependent details.

### 1 Purpose

S30NCF computes the European option price given by Heston's stochastic volatility model with term structure.

### 2 Specification

```

SUBROUTINE S30NCF (CALPUT, M, NUMTS, X, FWD, DISC, TS, T, ALPHA, LAMBDA,      &
                  CORR, SIGMAT, VARO, P, IFAIL)
INTEGER           M, NUMTS, IFAIL
REAL (KIND=nag_wp) X(M), FWD, DISC, TS(NUMTS), T, ALPHA(NUMTS),          &
                  LAMBDA(NUMTS), CORR(NUMTS), SIGMAT(NUMTS), VARO,      &
                  P(M)
CHARACTER(1)     CALPUT

```

### 3 Description

S30NCF computes the price of a European option for Heston's stochastic volatility model with time-dependent parameters which are piecewise constant. Starting from the stochastic volatility model given by the Stochastic Differential Equation (SDE) system defined by Heston (1993), a scaling of the variance process is introduced, together with a normalization, setting the long run variance,  $\eta$ , equal to 1. This leads to

$$\frac{dS_t}{S_t} = \mu_t dt + \sigma_t \sqrt{\nu_t} dW_t^{(1)}, \quad (1)$$

$$d\nu_t = \lambda_t(1 - \nu_t)dt + \alpha_t \sqrt{\nu_t} dW_t^{(2)}, \quad (2)$$

$$\text{Cov} [dW_t^{(1)}, dW_t^{(2)}] = \rho_t dt, \quad (3)$$

where  $\mu_t = r_t - q_t$  is the drift term representing the contribution of interest rates,  $r_t$ , and dividends,  $q_t$ , while  $\sigma_t$  is the scaling parameter,  $\nu_t$  is the scaled variance,  $\lambda_t$  is the mean reversion rate and  $\alpha_t$  is the volatility of the scaled volatility,  $\sqrt{\nu_t}$ . Then,  $W_t^{(i)}$ , for  $i = 1, 2$ , are two standard Brownian motions with correlation parameter  $\rho_t$ . Without loss of generality, the drift term,  $\mu_t$ , is eliminated by modelling the forward price,  $F_t$ , directly, instead of the spot price,  $S_t$ , with

$$F_t = S_0 \exp\left(\int_0^t \mu_s ds\right). \quad (4)$$

If required, the spot can be expressed as,  $S_0 = DF_t$ , where  $D$  is the discount factor.

The option price is computed by dividing the time to expiry,  $T$ , into  $n_s$  subintervals  $[t_0, t_1], \dots, [t_{i-1}, t_i], \dots, [t_{n_s-1}, T]$  and applying the method of characteristic functions to each subinterval, with appropriate initial conditions. Thus, a pair of ordinary differential equations (one of which is a Riccati equation) is solved on each subinterval as outlined in Elices (2008) and Mikhailov and Nîgel (2003). Reversing time by taking  $\tau = T - t$ , the characteristic function solution for the first time subinterval, starting at  $\tau = 0$ , is given by Heston (1993), while the solution on each following subinterval uses the solution of the preceding subinterval as initial condition to compute the value of the characteristic function.

In the case of a ‘flat’ term structure, i.e., the parameters are constant over the time of the option,  $T$ , the form of the SDE system given by Heston (1993) can be recovered by setting  $\kappa = \lambda_t$ ,  $\eta = \sigma_t^2$ ,  $\sigma_v = \sigma_t \alpha_t$  and  $V_0 = \sigma_t^2 V_0$ .

Conversely, given the Heston form of the SDE pair, to get the term structure form set  $\lambda_t = \kappa$ ,  $\sigma_t = \sqrt{\eta}$ ,  $\alpha_t = \frac{\sigma_v}{\sqrt{\eta}}$  and  $V_0 = \frac{V_0}{\eta}$ .

## 4 References

Bain A (2011) *Private communication*

Elices A (2008) Models with time-dependent parameters using transform methods: application to Heston's model *arXiv:0708.2020v2*

Heston S (1993) A closed-form solution for options with stochastic volatility with applications to bond and currency options *Review of Financial Studies* **6** 327–343

Mikhailov S and Nígel U (2003) Heston's Stochastic Volatility Model Implementation, Calibration and Some Extensions *Wilmott Magazine July/August* 74–79

## 5 Arguments

- 1: CALPUT – CHARACTER(1) *Input*  
*On entry:* determines whether the option is a call or a put.  
 CALPUT = 'C'  
     A call; the holder has a right to buy.  
 CALPUT = 'P'  
     A put; the holder has a right to sell.  
*Constraint:* CALPUT = 'C' or 'P'.
- 2: M – INTEGER *Input*  
*On entry:*  $m$ , the number of strike prices to be used.  
*Constraint:*  $M \geq 1$ .
- 3: NUMTS – INTEGER *Input*  
*On entry:*  $n_s$ , the number of subintervals into which the time to expiry,  $T$ , is divided.  
*Constraint:* NUMTS  $\geq 1$ .
- 4: X(M) – REAL (KIND=nag\_wp) array *Input*  
*On entry:*  $X(i)$  contains the  $i$ th strike price, for  $i = 1, 2, \dots, m$ .  
*Constraint:*  $X(i) \geq z$  and  $X(i) \leq 1/z$ , where  $z = X02AMF()$ , the safe range parameter, for  $i = 1, 2, \dots, M$ .
- 5: FWD – REAL (KIND=nag\_wp) *Input*  
*On entry:* the forward price of the asset.  
*Constraint:*  $FWD \geq z$  and  $FWD \leq 1/z$ , where  $z = X02AMF()$ , the safe range parameter.
- 6: DISC – REAL (KIND=nag\_wp) *Input*  
*On entry:* the discount factor, where the current price of the underlying asset,  $S_0$ , is given as  $S_0 = DISC \times FWD$ .  
*Constraint:*  $DISC \geq z$  and  $DISC \leq 1/z$ , where  $z = X02AMF()$ , the safe range parameter.

- 7: TS(NUMTS) – REAL (KIND=nag\_wp) array Input  
*On entry:* TS( $i$ ) must contain the length of the time intervals for which the corresponding element of ALPHA, LAMBDA, CORR and SIGMAT apply. These should be ordered as they occur in time i.e.,  $\Delta t_i = t_i - t_{i-1}$ .  
*Constraint:* TS( $i$ )  $\geq z$  and TS( $i$ )  $\leq 1/z$ , where  $z = X02AMF()$ , the safe range parameter, for  $i = 1, 2, \dots, \text{NUMTS}$ .
- 8: T – REAL (KIND=nag\_wp) Input  
*On entry:* T contains the time to expiry. If  $T > \sum \Delta t_i$  then the parameters associated with the last time interval are extended to the expiry time. If  $T < \sum \Delta t_i$  then the parameters specified are used up until the expiry time. The rest are ignored.  
*Constraint:* T  $\geq z$ , where  $z = X02AMF()$ , the safe range parameter.
- 9: ALPHA(NUMTS) – REAL (KIND=nag\_wp) array Input  
*On entry:* ALPHA( $i$ ) must contain the value of  $\alpha_t$ , the value of the volatility of the scaled volatility,  $\sqrt{\nu}$ , over time subinterval  $\Delta t_i$ .  
*Constraint:* ALPHA( $i$ )  $\geq z$  and ALPHA( $i$ )  $\leq 1/z$ , where  $z = X02AMF()$ , the safe range parameter, for  $i = 1, 2, \dots, \text{NUMTS}$ .
- 10: LAMBDA(NUMTS) – REAL (KIND=nag\_wp) array Input  
*On entry:* LAMBDA( $i$ ) must contain the value,  $\lambda_t$ , of the mean reversion parameter over the time subinterval  $\Delta t_i$ .  
*Constraint:* LAMBDA( $i$ )  $\geq z$  and LAMBDA( $i$ )  $\leq 1/z$ , where  $z = X02AMF()$ , the safe range parameter, for  $i = 1, 2, \dots, \text{NUMTS}$ .
- 11: CORR(NUMTS) – REAL (KIND=nag\_wp) array Input  
*On entry:* CORR( $i$ ) must contain the value,  $\rho_t$ , of the correlation parameter over the time subinterval  $\Delta t_i$ .  
*Constraint:*  $-1.0 \leq \text{CORR}(i) \leq 1.0$ , for  $i = 1, 2, \dots, \text{NUMTS}$ .
- 12: SIGMAT(NUMTS) – REAL (KIND=nag\_wp) array Input  
*On entry:* SIGMAT( $i$ ) must contain the value,  $\sigma_t$ , of the variance scale factor over the time subinterval  $\Delta t_i$ .  
*Constraint:* SIGMAT( $i$ )  $\geq z$  and SIGMAT( $i$ )  $\leq 1/z$ , where  $z = X02AMF()$ , the safe range parameter, for  $i = 1, 2, \dots, \text{NUMTS}$ .
- 13: VAR0 – REAL (KIND=nag\_wp) Input  
*On entry:*  $\nu_0$ , the initial scaled variance.  
*Constraint:* VAR0  $\geq 0.0$ .
- 14: P(M) – REAL (KIND=nag\_wp) array Output  
*On exit:* P( $i$ ) contains the computed option price at the expiry time,  $T$ , corresponding to strike X( $i$ ) for the specified term structure, for  $i = 1, 2, \dots, M$ .
- 15: IFAIL – INTEGER Input/Output  
*On entry:* IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation for details.  
 For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then

the value 1 is recommended. Otherwise, if you are not familiar with this argument, the recommended value is 0. **When the value  $-1$  or  $1$  is used it is essential to test the value of IFAIL on exit.**

*On exit:* IFAIL = 0 unless the routine detects an error or a warning has been flagged (see Section 6).

## 6 Error Indicators and Warnings

If on entry IFAIL = 0 or  $-1$ , explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL = 1

On entry, CALPUT =  $\langle value \rangle$  was an illegal value.

IFAIL = 2

On entry, M =  $\langle value \rangle$ .  
Constraint:  $M \geq 1$ .

IFAIL = 3

On entry, NUMTS =  $\langle value \rangle$ .  
Constraint:  $NUMTS \geq 1$ .

IFAIL = 4

On entry, X( $\langle value \rangle$ ) =  $\langle value \rangle$ .  
Constraint:  $\langle value \rangle \leq X(i) \leq \langle value \rangle$ .

IFAIL = 5

On entry, FWD =  $\langle value \rangle$ .  
Constraint:  $\langle value \rangle \leq FWD \leq \langle value \rangle$ .

IFAIL = 6

On entry, DISC =  $\langle value \rangle$ .  
Constraint:  $\langle value \rangle \leq DISC \leq \langle value \rangle$ .

IFAIL = 7

On entry, TS( $\langle value \rangle$ ) =  $\langle value \rangle$ .  
Constraint:  $\langle value \rangle \leq TS(i) \leq \langle value \rangle$ .

IFAIL = 8

On entry, T =  $\langle value \rangle$ .  
Constraint:  $T \geq \langle value \rangle$ .

IFAIL = 9

On entry, ALPHA( $\langle value \rangle$ ) =  $\langle value \rangle$ .  
Constraint:  $\langle value \rangle \leq ALPHA(i) \leq \langle value \rangle$ .

IFAIL = 10

On entry, LAMBDA( $\langle value \rangle$ ) =  $\langle value \rangle$ .  
Constraint:  $\langle value \rangle \leq LAMBDA(i) \leq \langle value \rangle$ .

IFAIL = 11

On entry, CORR(*value*) = *value*.  
Constraint:  $|\text{CORR}(i)| \leq 1.0$ .

IFAIL = 12

On entry, SIGMAT(*value*) = *value*.  
Constraint:  $\langle \text{value} \rangle \leq \text{SIGMAT}(i) \leq \langle \text{value} \rangle$ .

IFAIL = 13

On entry, VAR0 = *value*.  
Constraint:  $\text{VAR0} > 0.0$ .

IFAIL = 14

Quadrature has not converged to the specified accuracy. However, the result should be a reasonable approximation.

IFAIL = 15

Solution cannot be computed accurately. Check values of input arguments.

IFAIL = -99

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL = -399

Your licence key may have expired or may not have been installed correctly.

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL = -999

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

## 7 Accuracy

The solution is obtained by integrating the pair of ordinary differential equations over each subinterval in time. The accuracy is controlled by a relative tolerance over each time subinterval, which is set to  $10^{-8}$ . Over a number of subintervals in time the error may accumulate and so the overall error in the computation may be greater than this. A threshold of  $10^{-10}$  is used and solutions smaller than this are not accurately evaluated.

## 8 Parallelism and Performance

S30NCF is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

## 9 Further Comments

None.

## 10 Example

This example computes the price of a European call using Heston's stochastic volatility model with a term structure of interest rates.

### 10.1 Program Text

```

Program s30ncfe

!      S30NCF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
Use nag_library, Only: s30ncf
Use nag_precisions, Only: wp
!      .. Implicit None Statement ..
Implicit None
!      .. Parameters ..
Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
Real (Kind=wp)              :: disc, fwd, t, var0
Integer                     :: i, ifail, m, numts
Character (1)               :: calput
!      .. Local Arrays ..
Real (Kind=wp), Allocatable :: alpha(:), corr(:), lambda(:), p(:), &
                             sigmat(:), ts(:), x(:)

!      .. Executable Statements ..
Write (nout,*) 'S30NCF Example Program Results'

!      Skip heading in data file

Read (nin,*)

Read (nin,*) calput
Read (nin,*) m, numts

Allocate (p(m),ts(numts),x(m),alpha(numts),corr(numts),lambda(numts), &
         sigmat(numts))

Read (nin,*) fwd, disc, var0
Read (nin,*) x(1:m)
Read (nin,*) ts(1:numts)
Read (nin,*) t
Read (nin,*) alpha(1:numts)
Read (nin,*) corr(1:numts)
Read (nin,*) lambda(1:numts)
Read (nin,*) sigmat(1:numts)

ifail = 0
Call s30ncf(calput,m,numts,x,fwd,disc,ts,t,alpha,lambda,corr,sigmat, &
         var0,p,ifail)

If (ifail/=0) Then
  Go To 100
End If

Write (nout,*)
Write (nout,*) &
  'Heston''s Stochastic volatility Model with Term Structure'

Select Case (calput)
Case ('C','c')
  Write (nout,*) 'European Call :'
Case ('P','p')
  Write (nout,*) 'European Put :'
End Select

Write (nout,99998) ' Forward = ', fwd

```

```

Write (nout,99998) ' Discount Factor          = ', disc
Write (nout,99998) ' Variance                = ', var0

Write (nout,*) '   ts          alpha    lambda    corr    sigmat'
Do i = 1, numts
  Write (nout,99997) ts(i), alpha(i), lambda(i), corr(i), sigmat(i)
End Do

Write (nout,*)
Write (nout,*) '   Strike    Expiry      Option Price'

Do i = 1, m
  Write (nout,99999) x(i), t, p(i)
End Do

100 Continue

99999 Format (1X,2(F9.4,1X),3X,F9.4)
99998 Format (A,1X,F8.4)
99997 Format (1X,5(F9.4,1X))
End Program s30ncfe

```

## 10.2 Program Data

S30NCF Example Program Data

'C'	: Call = 'C', Put = 'P'
1 2	: M, NUMTS
100.0 1.0 1.0	: FWD, DISC, VARO
100.0	: X(I), I = 1,2,...M
0.35 0.65	: TS(I), I = 1,2,...NUMTS
1.0	: T
2.25 1.5	: ALPHA(I), I = 1,2,...NUMTS
-0.05 0.1	: CORR(I), I = 1,2,...NUMTS
2.0 1.5	: LAMBDA(I), I = 1,2,...NUMTS
0.04 0.13	: SIGMAT(I), I = 1,2,...NUMTS

## 10.3 Program Results

S30NCF Example Program Results

Heston's Stochastic volatility Model with Term Structure  
European Call :

Forward	=	100.0000		
Discount Factor	=	1.0000		
Variance	=	1.0000		
ts	alpha	lambda	corr	sigmat
0.3500	2.2500	2.0000	-0.0500	0.0400
0.6500	1.5000	1.5000	0.1000	0.1300
Strike	Expiry	Option Price		
100.0000	1.0000	4.0074		

---