# NAG Library Routine Document

# S30BAF

**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

S30BAF computes the price of a floating-strike lookback option.

## 2 Specification

```
SUBROUTINE S30BAF(CALPUT, M, N, SM, S, T, SIGMA, R, Q, P, LDP, IFAIL)
INTEGER           M, N, LDP, IFAIL
double precision  SM(M), S, T(N), SIGMA, R, Q, P(LDP,N)
CHARACTER*1       CALPUT
```

## 3 Description

S30BAF computes the price of a floating-strike lookback call or put option. A call option of this type confers the right to buy the underlying asset at the lowest price, $S_{\min}$, observed during the lifetime of the contract. A put option gives the holder the right to sell the underlying asset at the maximum price, $S_{\max}$, observed during the lifetime of the contract. Thus, at expiry, the payoff for a call option is $S - S_{\min}$, and for a put, $S_{\max} - S$.

For a given minimum value the price of a floating-strike lookback call with underlying asset price, $S$, and time to expiry, $T$, is

$$P_{\text{call}} = Se^{-qT}\Phi(a_1) - S_{\min}e^{-rT}\Phi(a_2) + Se^{-rT}\frac{\sigma^2}{2b}\left[\left(\frac{S}{S_{\min}}\right)^{-2b/\sigma^2}\Phi\left(-a_1 + \frac{2b}{\sigma}\sqrt{T}\right) - e^{bT}\Phi(-a_1)\right],$$

where $b = r - q \neq 0$. The volatility, $\sigma$, risk-free interest rate, $r$, and annualised dividend yield, $q$, are constants. When $r = q$, the option price is given by

$$P_{\text{call}} = Se^{-qT}\Phi(a_1) - S_{\min}e^{-rT}\Phi(a_2) + Se^{-rT}\sigma\sqrt{T}[\phi(a_1) + a_1(\Phi(a_1) - 1)].$$

The corresponding put price is (for $b \neq 0$),

$$P_{\text{put}} = S_{\max}e^{-rT}\Phi(-a_2) - Se^{-qT}\Phi(-a_1) + Se^{-rT}\frac{\sigma^2}{2b}\left[-\left(\frac{S}{S_{\max}}\right)^{-2b/\sigma^2}\Phi\left(a_1 - \frac{2b}{\sigma}\sqrt{T}\right) + e^{bT}\Phi(a_1)\right].$$

When $r = q$,

$$P_{\text{put}} = S_{\max}e^{-rT}\Phi(-a_2) - Se^{-qT}\Phi(-a_1) + Se^{-rT}\sigma\sqrt{T}[\phi(a_1) + a_1\Phi(a_1)].$$

In the above, $\Phi$ denotes the cumulative Normal distribution function,

$$\Phi(x) = \int_{-\infty}^{x}\phi(y)dy$$

where $\phi$ denotes the standard Normal probability density function

$$\phi(y) = \frac{1}{\sqrt{2\pi}}\exp\left(-y^2/2\right)$$

and

$$a_1 = \frac{\ln(S/S_m)+\left(b+\sigma^2/2\right)T}{\sigma\sqrt{T}}$$

$$a_2 = a_1 - \sigma\sqrt{T}$$

where $S_m$ is taken to be the minimum price attained by the underlying asset, $S_{\min}$, for a call and the maximum price, $S_{\max}$, for a put.

## 4 References

Goldman B M, Sosin H B and Gatto M A (1979) Path dependent options: buy at the low, sell at the high *Journal of Finance* **34** 1111–1127

## 5 Parameters

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*: the number of minimum or maximum prices to be used.

*Constraint*: M $\geq 1$.

3: N – INTEGER *Input*

*On entry*: the number of times to expiry to be used.

*Constraint*: N $\geq 1$.

4: SM(M) – ***double precision*** array *Input*

*On entry*: SM($i$) must contain $S_{\min}(i)$, the $i$th minimum observed price of the underlying asset when CALPUT = 'C', or $S_{\max}(i)$, the maximum observed price when CALPUT = 'P', for $i = 1, 2, \ldots, \mathrm{M}$.

*Constraints*:

SM($i$) $\geq z$ and SM($i$) $\leq 1/z$, where $z =$ X02AMF(), the safe range parameter, for $i = 1, 2, \ldots, \mathrm{M}$;
if CALPUT = 'C', SM($i$) $\leq S$, for $i = 1, 2, \ldots, \mathrm{M}$;
if CALPUT = 'P', SM($i$) $\geq S$, for $i = 1, 2, \ldots, \mathrm{M}$.

5: S – ***double precision*** *Input*

*On entry*: $S$, the price of the underlying asset.

*Constraint*: S $\geq z$ and S $\leq 1/z$, where $z =$ X02AMF(), the safe range parameter.

6: T(N) – ***double precision*** array *Input*

*On entry*: T($i$) must contain $T_i$, the $i$th time, in years, to expiry, for $i = 1, 2, \ldots, \mathrm{N}$.

*Constraint*: T($i$) $\geq z$, where $z =$ X02AMF(), the safe range parameter, for $i = 1, 2, \ldots, \mathrm{N}$.

7: SIGMA – ***double precision*** *Input*

*On entry*: $\sigma$, the volatility of the underlying asset. Note that a rate of 15% should be entered as 0.15.

*Constraint*: SIGMA $> 0.0$.

8: R – ***double precision*** *Input*

*On entry*: $r$, the annual risk-free interest rate, continuously compounded. Note that a rate of 5% should be entered as 0.05.

*Constraint*: R $\geq 0.0$.

9: Q – ***double precision*** *Input*

*On entry*: $q$, the annual continuous yield rate. Note that a rate of 8% should be entered as 0.08.

*Constraint*: Q $\geq 0.0$.

10: P(LDP,N) – ***double precision*** array *Output*

*On exit*: the $m \times n$ array P contains the computed option prices.

11: LDP – INTEGER *Input*

*On entry*: the first dimension of the array P as declared in the (sub)program from which S30BAF is called.

*Constraint*: LDP $\geq$ M.

12: IFAIL – INTEGER *Input/Output*

*On entry*: IFAIL must be set to $0$, $-1$ or $1$. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

*On exit*: IFAIL $= 0$ unless the routine detects an error (see Section 6).

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 parameter, the recommended value is $0$. **When the value $-1$ or $1$ is used it is essential to test the value of IFAIL on exit.**

## 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 $\neq$ 'C' or 'P'.

IFAIL $= 2$

On entry, M $\leq 0$.

IFAIL $= 3$

On entry, N $\leq 0$.

IFAIL $= 4$

On entry, SM$(i) < z$ or SM$(i) > 1/z$, where $z =$ X02AMF(), the safe range parameter,
or CALPUT = 'C' and SM$(i) > S$,
or CALPUT = 'P' and SM$(i) < S$.

IFAIL $= 5$

On entry, S $< z$ or S $> 1/z$, where $z =$ X02AMF(), the safe range parameter.

IFAIL $= 6$

On entry, T$(i) < z$, where $z =$ X02AMF(), the safe range parameter.

IFAIL $= 7$

On entry, SIGMA $\leq 0.0$.

IFAIL $= 8$

On entry, R $< 0.0$.

IFAIL $= 9$

On entry, Q $< 0.0$.

IFAIL $= 11$

On entry, LDP $<$ M.

## 7 Accuracy

The accuracy of the output is dependent on the accuracy of the cumulative Normal distribution function, $\Phi$. This is evaluated using a rational Chebyshev expansion, chosen so that the maximum relative error in the expansion is of the order of the ***machine precision*** (see S15ABF and S15ADF). An accuracy close to ***machine precision*** can generally be expected.

## 8 Further Comments

None.

## 9 Example

This example computes the price of a floating-strike lookback call with a time to expiry of 6 months and a stock price of 120. The minimum price observed so far is 100. The risk-free interest rate is 10% per year and the volatility is 30% per year with an annual dividend return of 6%.

### 9.1 Program Text

```
*     S30BAF Example Program Text
*     Mark 22 Release. NAG Copyright 2007.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          LDP, MMAX, NMAX
      PARAMETER        (LDP=10,MMAX=10,NMAX=10)
*     .. Local Scalars ..
      DOUBLE PRECISION Q, R, S, SIGMA
      INTEGER          I, IFAIL, J, M, N
      CHARACTER        PUT
*     .. Local Arrays ..
      DOUBLE PRECISION P(LDP,NMAX), SM(MMAX), T(NMAX)
*     .. External Subroutines ..
      EXTERNAL         S30BAF
```

```
*     .. Executable Statements ..
      WRITE (NOUT,*) 'S30BAF Example Program Results'
      WRITE (NOUT,*)
      WRITE (NOUT,*) 'Floating-strike Lookback'
*     Skip heading in data file
      READ (NIN,*)
*     Read Problem parameters.
      READ (NIN,*) PUT
      READ (NIN,*) S, SIGMA, R, Q
      READ (NIN,*) M, N
*
      IF (M.LE.MMAX .AND. N.LE.NMAX) THEN
*        Read array of MIN/MAX prices, SM
         READ (NIN,*) (SM(I),I=1,M)
*        Read array of times to expiry
         READ (NIN,*) (T(I),I=1,N)
*
         IFAIL = 1
*
         CALL S30BAF(PUT,M,N,SM,S,T,SIGMA,R,Q,P,LDP,IFAIL)
*
         IF (IFAIL.EQ.0) THEN
            IF (PUT.EQ.'C') THEN
               WRITE (NOUT,*) 'European Call :'
            ELSE IF (PUT.EQ.'P') THEN
               WRITE (NOUT,*) 'European Put :'
            END IF
            WRITE (NOUT,'(A,1X,F8.4)') '  Spot       = ', S
            WRITE (NOUT,'(A,1X,F8.4)') '  Volatility = ', SIGMA
            WRITE (NOUT,'(A,1X,F8.4)') '  Rate       = ', R
            WRITE (NOUT,'(A,1X,F8.4)') '  Dividend   = ', Q
*
            WRITE (NOUT,*)
            WRITE (NOUT,*) '   Strike    Expiry    Option Price'
            DO 40 I = 1, M
               DO 20 J = 1, N
                  WRITE (NOUT,99999) SM(I), T(J), P(I,J)
   20          CONTINUE
   40       CONTINUE
         ELSE
            WRITE (NOUT,*)
            WRITE (NOUT,99998) IFAIL
         END IF
      END IF
*
99999 FORMAT (1X,2(F9.4,1X),6X,F9.4)
99998 FORMAT (1X,' ** S30BAF returned with IFAIL = ',I5)
      END
```

## 9.2 Program Data

```
S30BAF Example Program Data
 'C'                  : Call = 'C', Put = 'P'
 120.0 0.3 0.1 0.06   : S, SIGMA, R, Q
 1 1                  : M, N
 100.0                : SM(I), I = 1,2,...M
 0.5                  : T(I), I = 1,2,...N
```

## 9.3 Program Results

```
 S30BAF Example Program Results

 Floating-strike Lookback
 European Call :
  Spot       =  120.0000
  Volatility =    0.3000
```

```
  Rate       =    0.1000
  Dividend   =    0.0600

    Strike    Expiry    Option Price
  100.0000    0.5000         25.3534
```

---