THE COMPLETE GUIDE TO
Option Pricing Formulas

SECOND EDITION

ESPEN GAARDER HAUG

McGraw-Hill
New York Chicago San Francisco
Lisbon London Madrid
Mexico City Milan Seoul New Delhi San Juan
Singapore Sydney Toronto

```
    Dim ht As Double, i As Double
    Dim Alpha As Double, Beta As Double

    If b >= r Then '// Never optimal to exersice before maturity
            BSAmericanCallApprox = GBlackScholes("c", S, X, T, r, b, v)
    Else
        Beta = (1 / 2 - b / v^2) + Sqr((b / v^2 - 1 / 2)^2 + 2 * r / v^2)
        BInfinity = Beta / (Beta - 1) * X
        B0 = Max(X, r / (r - b) * X)
        ht = -(b * T + 2 * v * Sqr(T)) * B0 / (BInfinity - B0)
        i = B0 + (BInfinity - B0) * (1 - Exp(ht))
        Alpha = (i - X) * i^(-Beta)
        If S >= i Then
            BSAmericanCallApprox = S - X
        Else
            BSAmericanCallApprox = Alpha * S ^ Beta _
      - Alpha * phi(S, T, Beta, i, i, r, b, v) _
     + phi(S, T, 1, i, i, r, b, v) - phi(S, T, 1, X, i, r, b, v) _
     - X * phi(S, T, 0, i, i, r, b, v) + X * phi(S, T, 0, X, i, r, b, v)
        End If
    End If

End Function


Function phi(S As Double, T As Double, gamma As Double, h As Double, i As Double
        r As Double, b As Double, v As Double) As Double

    Dim lambda As Double, kappa As Double
    Dim d As Double

    lambda = (-r + gamma * b + 0.5 * gamma * (gamma - 1) * v^2) * T
    d = -(Log(S/h) + (b + (gamma - 0.5) * v^2) * T) / (v * Sqr(T))
    kappa = 2 * b / (v^2) + (2 * gamma - 1)
    phi = Exp(lambda) * S^gamma * (CND(d) - (i / S)^kappa _
    * CND(d - 2 * Log(i / S) / (v * Sqr(T))))

End Function
```

where *CND*(·) is the cumulative normal distribution function described in Appendix A at the end of Chapter 1. Example: *BSAmericanApprox*("c", 42, 40, 0.75, 0.04, −0.04, 0.35) returns an American call value of 5.2704 as in the numerical example above.

3.3 THE BJERKSUND AND STENSLAND (2002) APPROXIMATION

The Bjerksund and Stensland (2002) approximation divides the time to maturity into two parts, each with a separate flat exercise boundary. It is thus a straightforward generalization of the Bjerksund-Stensland 1993 algorithm. The method is fast and efficient and should be more accurate than the Barone-Adesi and Whaley (1987) and the Bjerksund and Stensland (1993b) approximations. The algorithm requires an accurate cumulative bivariate normal approximation.

Several approximations that are described in the literature are not sufficiently accurate, but the Genze algorithm presented in Chapter 13 should do.

$$\begin{aligned}
C &= \alpha_2 S^{\beta} - \alpha_2 \phi(S, t_1, \beta, I_2, I_2) \\
&\quad + \phi(S, t_1, 1, I_2, I_2) - \phi(S, t_1, 1, I_1, I_2) \\
&\quad - X\phi(S, t_1, 0, I_2, I_2) + X\phi(S, t_1, 0, I_1, I_2) \\
&\quad + \alpha_1 \phi(S, t_1, \beta, I_1, I_2) - \alpha_1 \Psi(S, T, \beta, I_1, I_2, I_1, t_1) \\
&\quad + \Psi(S, T, 1, I_1, I_2, I_1, t_1) - \Psi(S, T, 1, X, I_2, I_1, t_1) \\
&\quad - X\Psi(S, T, 0, I_1, I_2, I_1, t_1) + \Psi(S, T, 0, X, I_2, I_1, t_1),
\end{aligned} \tag{3.2}$$

where

$$\alpha_1 = (I_1 - X) I_1^{-\beta}, \quad \alpha_2 = (I_2 - X) I_2^{-\beta}$$

$$\beta = \left(\frac{1}{2} - \frac{b}{\sigma^2}\right) + \sqrt{\left(\frac{b}{\sigma^2} - \frac{1}{2}\right)^2 + 2\frac{r}{\sigma^2}}$$

The function $\phi(S, T, \gamma, H, I)$ is given by

$$\phi(S, T, \gamma, H, I) = e^{\lambda} S^{\gamma} \left[N(-d) - \left(\frac{I}{S}\right)^{\kappa} N(-d_2) \right]$$

$$d = \frac{\ln(S/H) + [b + (\gamma - \frac{1}{2})\sigma^2]T}{\sigma\sqrt{T}}$$

$$d_2 = \frac{\ln(I^2/(SH)) + [b + (\gamma - \frac{1}{2})\sigma^2]T}{\sigma\sqrt{T}}$$

$$\lambda = -r + \gamma b + \frac{1}{2}\gamma(\gamma - 1)\sigma^2$$

$$\kappa = \frac{2b}{\sigma^2} + (2\gamma - 1),$$

and the trigger price I is defined as

$$I_1 = B_0 + (B_\infty - B_0)(1 - e^{h_1})$$

$$I_2 = B_0 + (B_\infty - B_0)(1 - e^{h_2})$$

$$h_1 = -(bt_1 + 2\sigma\sqrt{t_1}) \left(\frac{X^2}{(B_\infty - B_0)B_0}\right)$$

$$h_2 = -(bT + 2\sigma\sqrt{T}) \left(\frac{X^2}{(B_\infty - B_0)B_0}\right)$$

$$t_1 = \frac{1}{2}(\sqrt{5} - 1)T$$

$$B_\infty = \frac{\beta}{\beta - 1}X$$

$$B_0 = \max\left[X, \left(\frac{r}{r - b}\right)X\right]$$

Moreover, the function $\Psi(S, T, \gamma, H, I_2, I_1, t_1)$ is given by

$$\Psi(S, T, \gamma, H, I_2, I_1, t_1, r, b, \sigma) = e^{\lambda T} S^\gamma \left[M(-e_1, -f_1, \rho) - (I_2/S)^\kappa M(-e_2, -f_2, \rho) - (I_1/S)^\kappa M(-e_3, -f_3, -\rho) + (I_1/I_2)^\kappa M(-e_4, -f_4, -\rho)\right],$$

where

$$e_1 = \frac{\ln(S/I_1) + (b + (\gamma - \frac{1}{2})\sigma^2)t_1}{\sigma\sqrt{t_1}} \qquad e_2 = \frac{\ln(I_2^2/(SI_1)) + (b + (\gamma - \frac{1}{2})\sigma^2)t_1}{\sigma\sqrt{t_1}}$$

$$e_3 = \frac{\ln(S/I_1) - (b + (\gamma - \frac{1}{2})\sigma^2)t_1}{\sigma\sqrt{T}} \qquad e_4 = \frac{\ln(I_2^2/(SI_1)) - (b + (\gamma - \frac{1}{2})\sigma^2)t_1}{\sigma\sqrt{t_1}}$$

$$f_1 = \frac{\ln(S/H) + (b + (\gamma - \frac{1}{2})\sigma^2)T}{\sigma\sqrt{T}} \qquad f_2 = \frac{\ln(I_2^2/(SH)) + (b + (\gamma - \frac{1}{2})\sigma^2)T}{\sigma\sqrt{T}}$$

$$f_3 = \frac{\ln(I_2^2/(SH)) + (b + (\gamma - \frac{1}{2})\sigma^2)T}{\sigma\sqrt{T}} \qquad f_4 = \frac{\ln(SI_1^2/(HI_2^2)) + (b + (\gamma - \frac{1}{2})\sigma^2)T}{\sigma\sqrt{T}}$$

Table 3-2 gives numerical values for the Bjerksund-Stensland (2002) approximation and also their 1993 approximation. The 2002 version of their model is more accurate but slightly more computer-intensive.

Computer algorithm

The computer code for the Bjerksund and Stensland 2002 American option approximation consists of four functions. The first one checks if the option is a call or a put. If the option is a put, the function uses the American put-call transformation. The function then calls the main function *BSAmericanCallApprox*2002$(\cdot)$, which calculates the option value. The main function uses two other functions: the $phi(\cdot)$ function, which in the formula above is described as $\phi(S, T, \gamma, H, I)$ and the $ksi(\cdot)$ function described as $\Psi(S, T, \gamma, H, I_2, I_1, t_1, r, b, \sigma)$, and the *GBlackScholes*$(\cdot)$ function, which is the generalized BSM formula described in Chapter 1.

```
Function BSAmericanApprox2002(CallPutFlag As String, S As Double, X As Double, _
T As Double, r As Double, b As Double, v As Double) As Double

    If CallPutFlag = "c" Then
        BSAmericanApprox2002 = BSAmericanCallApprox2002(S, X, T, r, b, v)
```