

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] BAIDYA, T. K. N; CASTRO, A. L. **Convergência dos Modelos de árvores binomiais para avaliação de opções. Pesquisa Operacional.** Vol. 21 pp.17-30, junho 2001.
- [2] BARBE, T. **Aplicações de Quase Monte – Carlo no Mercado de derivativos brasileiro.** Dissertação de Mestrado, Departamento de Economia, USP. São Paulo 2001.
- [3] BRANDIMARTE, P. **Numerical Methods in Finance.** New York: John Wiley & sons, 2002.
- [4] BROADIE, M.; GLASSERMAN, P.: **Pricing American-Style Securities Using Simulation, Journal of Economic Dynamics and Control** , June 1997, vol.21, nº 8-9, pp.1323-1352.
- [5] BROADIE, M.; GLASSERMAN, P.; JAIN, G. **Enhanced Monte Carlo Estimates for American Option Prices, Journal of Derivatives**, vol.5, pp.25-44.
- [6] CASTRO, A.L. **Avaliação de Investimento de Capital em Projetos de Geração Termoelétrica no Setor Elétrico Brasileiro Usando Teoria das Opções Reais**, Dissertação de Mestrado, Departamento de Engenharia Industrial, PUC-Rio, Rio de Janeiro, 2000.
- [7] CHRISS, N. A. **Black – Scholes and beyond option pricing models.** Chicago: Irwin Professional Publishing, 1997.
- [8] CLEWTOW, L.; STRIKLAND, C. **Implementing derivatives models.** New York: John Wiley & sons, 1998.
- [9] DIAS, M.A.G.; **Investimento Sob Incerteza em Exploração e Produção de Petróleo**, Dissertação de Mestrado, Departamento de Engenharia Industrial, PUC-Rio, 1996.
- [10] DUAN, J. ; SIMONATO, J. **American Option Pricing under GARCH by a Markov Chain Approximation**, Journal of Economic Dynamics & Control n.25, 2001 pp 1689-1718.

- [11] DUAN, J. ; SIMONATO, J; GAUTHIER, G; SASSEVILLE, C. **Aproximating American Option Prices in the GARCH Framework**, Working paper to appear in Journal of Futures Markets, January, 2003.
- [12] DUPIRE, B. (Ed). **Monte Carlo – Methodologies and Applications for Pricing and Risk Management**. Risk Books, 1998.
- [13] ENGLE, R. **Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of UK Inflation**, *Econometrica*, n 50 (1983), 987-1108.
- [14] ENGLE, R; V. NG. **Measuring and Testing of the Impact of News on Volatility**, *Journal of Finance* 48 (1993), 1749-1778.
- [15] FORSYTH, P. A. **An Introduction to Computational Finance Without Agonizing Pain**. University of Waterloo Working Paper. Ontario, June 23, 2002.
- [16] FU, M.C.; LAPRISE, S.B.; MADAN, D.B.; SU Y.; WU, R. **Pricing American Options: A Comparison of Monte Carlo Simulation Approaches**. Working Paper, University of Maryland at College Park, April 2000, 44 pp., and *Journal of Computational Finance*, Vol.4, nº 3, Spring 2001, pp.39-88.
- [17] GARCIA, D. **A Monte Carlo Method for pricing American Options**. Working Paper, University of California at Berkeley, January 2000, 43 pp
- [18] GRANT, D.; VORA, G.; WEEKS, D.E. **Path-Dependent Options: Extending the Monte Carlo Simulation Approach**. *Management Science*, vol.43, no 11, November 1997, pp.1589-1602.
- [19] GRANT, D.; VORA, G.; WEEKS, D.E. **Simulation and Early-Exercise of Option Problem**. *Journal of Financial Engineering*, vol.5, nº 3, September 1996, pp.211-227.
- [20] HANSELMAN, D.; LITTLEFIELD, B. **Mastering Mat lab 6 – A Comprehensive Tutorial and Reference**. New York: Prentice Hall, 2001.
- [21] JACKEL, P. **Monte Carlo Methods in Finance**. New York: John Wiley & sons, 2002.
- [22] HAUG, E. G. **The Complete Guide to Option Pricing Formulas**. New York: Mc Graw Hill, 1998.

- [23] HOFFMANN, C. **Valuation of American Options**. Thesis submitted for the degree of Diploma in Mathematical Finance – University of Oxford, October 8, 2000
- [24] HULL, J. C. **Options, Futures & other Derivatives**. 4. ed. New York: Prentice Hall, 2000.
- [25] IBANEZ, A.; ZAPATERO, F. **Monte Carlo Valuation of American Options Through Computation of the Optimal Exercise Frontier**. Working Paper, Instituto Tecnológico Autónomo de México & University of South California, August 1999, 30 pp.
- [26] JOHNSTON, J.; DINARDO, J. **Métodos Econométricos**. 4. ed. Portugal : Mc Graw Hill, 2001.
- [27] JOY, C.; BOYLE, P.P.; TAN, K.S. **Quasi-Monte Carlo Methods in Numerical Finance**. Management Science, vol.42, no 6, June 1996, pgs.926-938.
- [28] JUNG, A. **Improving the Performance of Low-Discrepancy Sequences**. Journal of Derivatives, Winter 1998, pp.85-95.
- [29] KELLY, L.G. **Handbook of Numerical Methods and Applications**. California: Addison-Wesley Publishing Company, 1967.
- [30] KERMAN, J. **Numerical Methods for Option Pricing: Binomial and Finite-difference Approximations**. Courant Institute of Mathematical Sciences, New York University. January 15, 2002.
- [31] LEWIS, A. L. **Option Valuation under Stochastic Volatility with Mathematica code**. California: Finance Press, 2000.
- [32] LONGSTAFF, F.A.;SCHWARTZ, E.S. **Valuing American Options By Simulation: A Simple Least-Square Approach**. The Review of Financial Studies, Vol.14, nº1, Spring 2001, pp 113-147
- [33] MORO, B. **The Full Monte**. Risk. vol.8, no 2, February 1995.
- [34] ODEGAARD, B. A. **Financial Numerical Recipes**. [S.l.:s.n.] Working Paper, September 9, 1999
- [35] OKTEN, G. **Applications of a Hybrid-Monte Carlo Sequence to Option Pricing in Niederreiter & Spanier** (Eds.), Monte Carlo and Quasi-Monte Carlo Methods 1998 - Springer-Verlag, 2000, pp. 391-406.
- [36] OWEN, A. B. **Monte Carlo, Quasi-Monte Carlo and Randomized Quasi-Monte Carlo in Niederreiter & Spanier**. Monte Carlo and Quasi-

Monte Carlo Methods 1998 Springer-Verlag, 2000, Proceedings
Claremont Conference 1998, pp.86-97.

- [37] ROCHMAN, R. R. **Análise de Métodos numéricos para precificação de opções**, Dissertação de Mestrado, FGV/EAESP. São Paulo, 1998.
- [38] RUBINSTEIN, R. Y. **Simulation and The Monte Carlo Method**.]
York: John Wiley & sons, 1981.
- [39] TAN, K.S.; BOYLE, P.P. **Applications of Randomized Low Discrepancy Sequences to the Valuation of Complex Securities**.
Journal of Economic Dynamics & Control, vol.24, pp.1747-1782.
- [40] TAVELLA, D. A. **Quantitative Methods in Derivatives Pricing**. New
York: John Wiley & sons, 2002.
- [41] TAVELLA, D. A.; RANDALL, C. **Pricing Financial Instruments**. New
York: John Wiley & sons, 2002.
- [42] TIAN, T.; BURRAGE, K. **Accuracy Issues of Monte Carlo Methods for Valuing American Options**. Working Paper, The University of
Queens land, Australia.
- [43] Site de opções reais **[http: //www.puc-rio.br/marco.ind](http://www.puc-rio.br/marco.ind)**, último acesso
26/03/2003.
- [44] WILMOTT, P. **Derivatives – The theory and practice of financial engineering**. New York: John Wiley & sons, 1998.
- [45] WHITLOCK, K. **Monte Carlo Methods**. New York: John Wiley &
sons, 1986.1v.
- [46] XIAOQING and KIANGUAN, **A Parsimonious Monte Carlo Method for Pricing American-Style Options**, Research paper, Centre for
Financial Engineering, National University of Singapore, June, 2001.

Apêndice A:

DEFINIÇÕES

A.1. Processo de Wiener

Se uma variável modifica-se ao longo do tempo de maneira aleatória, dizemos que ela segue um processo estocástico que pode ser contínuo ou discreto. O processo de Markov é um processo estocástico onde o valor presente da variável é suficiente para determinarmos seu próximo valor. Assim, o próximo valor da variável independe do caminho de valores anteriores, mas apenas do seu valor final. Já o processo de Wiener é um processo de Markov com média 0 e variância 1. Em física, esse processo é comumente conhecido como *Brownian motion*. Se uma variável aleatória z segue um processo de Wiener, esta possui as seguintes propriedades:

- uma variação Δz durante um pequeno intervalo de tempo Δt é dado por:

$$\Delta z = \varepsilon\sqrt{\Delta t} \quad \text{ou fazendo } \Delta t \rightarrow 0 \text{ temos } dz = \varepsilon\sqrt{dt}$$

onde ε é uma variável aleatória sorteada de uma distribuição $N \sim (0,1)$.

Assim, Δz também segue uma distribuição normal $N \sim (0, \sqrt{\Delta t})$.

- Os valores de Δz para dois intervalos de tempo diferentes, devem ser independentes.

Uma variável x que segue um processo de Wiener generalizado, pode ser definido em termos de dz como:

$$dx = \alpha dt + \beta dz$$

onde α e b são constantes. O termo α representa o *drift* do processo e b sua variação. Já o termo diferencial dz é uma variável aleatória sorteada de uma distribuição $N \sim (0, \sqrt{dt})$. Os valores de dx para diferentes intervalos de tempo devem ser independentes.

Muitos instrumentos financeiros são representados por um processo onde os parâmetros α e b não são necessariamente constantes. Nesse caso, o processo estocástico passa a ser denominado processo de Itô.

$$dx = \alpha(x, t)dt + b(x, t)dz$$

Assim, considerando a variável S representativa do preço de um ativo, temos que seu processo estocástico pode ser definido como um passeio aleatório lognormal. Nesse caso, a equação acima assume os valores $\alpha(S, t) = \mu S$ e $b(S, t) = \sigma S$. Consequente, temos que:

$$dS = \mu Sdt + \sigma Sdz$$

onde μ (*drift*) e σ (volatilidade) são constantes.

Considerando a hipótese de neutralidade ao risco, a taxa de retorno ou *drift* (μ) da equação acima pode ser substituída pela taxa livre de risco.

$$dS = rSdt + \sigma Sdz$$

Assumindo que o ativo S paga uma taxa contínua de dividendos q , o *drift* deve então ser reduzido dessa mesma quantidade.

$$dS = (r - q)Sdt + \sigma Sdz$$

A.2. Lema de Itô

Suponha a seguinte função $V(x_t, t)$, onde o subscrito t indica que a variável x é uma função dependente do tempo. No cálculo diferencial temos a regra da cadeia, que nos permite derivar V em função do tempo.

$$dV = \frac{\partial V}{\partial x} dx + \frac{\partial V}{\partial t} dt$$

No entanto, quando temos funções envolvendo variáveis estocásticas, não podemos deriva-las aplicando diretamente a regra da cadeia conforme descrito acima. Assim, o Lema de Itô é o equivalente à regra da cadeia aplicada à uma variável estocástica x_t . Usando a expansão de Taylor e anulando os termos de ordem superior, a medida que $\Delta t \rightarrow 0$ temos que:

$$\begin{aligned} dV &= \frac{\partial V}{\partial t} dt + \frac{\partial V}{\partial x} dx + \frac{1}{2} \frac{\partial^2 V}{\partial x^2} dx^2 & \therefore & \quad dx = \alpha(x, t)dt + b(x, t)dz \\ &= \left(\frac{\partial V}{\partial t} + \frac{1}{2} \frac{\partial^2 V}{\partial x^2} \right) dt + \frac{\partial V}{\partial x} dx & & \quad dx^2 = dt \end{aligned}$$

Aplicando o Lema de Itô à função $V(S, t)$ dependente da variável preço do ativo S . Supondo que S segue um processo estocástico de Itô ($dS = \alpha(S, t)dt + b(S, t)dz$), temos:

$$dV = \left(\alpha(S, t) \frac{\partial V}{\partial t} + \frac{1}{2} b(S, t)^2 \frac{\partial^2 V}{\partial S^2} \right) dt + \frac{\partial V}{\partial S} dS$$

Uma aplicação bastante comum do Lema de Itô é o processo estocástico dado por $V(S) = \ln(S)$. Assim, temos que:

$$dV = \frac{\partial V}{\partial S} dS + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} dt$$

$$\begin{aligned}
&= \frac{1}{S} \left((r - q)Sdt + \sigma Sdz \right) - \frac{1}{2} \sigma^2 S^2 \frac{1}{S^2} dt \\
&= \left(r - q - \frac{\sigma^2}{2} \right) dt + \sigma dz \\
&= \left(r - q - \frac{\sigma^2}{2} \right) dt + \sigma dz
\end{aligned}$$

Trabalhando a equação acima:

$$d(\ln S) = \left(r - q - \frac{\sigma^2}{2} \right) dt + \sigma dz$$

$$\ln S_T - \ln S_0 = \left(r - q - \frac{\sigma^2}{2} \right) dt + \sigma dz$$

$$\ln \left(\frac{S_T}{S_0} \right) = \left(r - q - \frac{\sigma^2}{2} \right) dt + \sigma dz \quad \therefore \quad dz = \varepsilon \sqrt{T}$$

$$S_T = S_0 e^{\left(r - q - \frac{\sigma^2}{2} \right) dt + \sigma \varepsilon \sqrt{T}}$$

A.3. Dedução da Equação Diferencial de Black Scholes

Premissas: (1) preço S da ação segue um processo de Wiener generalizado; (2) *Short Selling* com total uso da receita de venda é permitido; (3) não existem custos de transação; (4) não há pagamento de dividendos durante a vida do derivativo; (5) não existem oportunidades de arbitragem sem risco; (6) negociação de ações é feita de modo contínuo; (6) a taxa de risco r é constante para todas os vencimentos.

Partimos da seguinte premissa para o preço da ação:

$$dS = \mu Sdt + \sigma Sdz$$

Seja f o preço de um derivativo dependente da ação básica S . É razoável supor que f seja uma função do preço da ação S e do tempo t . Então, pelo Lema de Itô, sabemos que a seguinte relação também é verdadeira:

$$df = \left(\frac{\partial f}{\partial S} \mu S + \frac{\partial f}{\partial t} + \frac{1}{2} \frac{\partial^2 f}{\partial S^2} \sigma^2 S^2 \right) dt + \frac{\partial f}{\partial S} \sigma S dz$$

As versões discretas das duas equações acima são:

$$\Delta S = \mu S \Delta t + \sigma S \Delta z$$

$$\Delta f = \left(\frac{\partial f}{\partial S} \mu S + \frac{\partial f}{\partial t} + \frac{1}{2} \frac{\partial^2 f}{\partial S^2} \sigma^2 S^2 \right) \Delta t + \frac{\partial f}{\partial S} \sigma S \Delta z$$

Note que o Δz das duas equações é o mesmo, e compõe a parte estocástica do modelo. Montando uma carteira apropriada de ações e opções, podemos eliminar este componente. Esta carteira é formado pela venda a descoberto de uma opção, de valor f , e pela compra de $\partial f / \partial S$ ações. Seja Π o valor desta carteira dado por:

$$\Pi = -f + \frac{\partial f}{\partial S} S$$

A variação $\Delta \Pi$ no valor desta carteira com o tempo Δt é dada por:

$$\Delta \Pi = -\Delta f + \frac{\partial f}{\partial S} \Delta S$$

Substituindo Δf e ΔS na equação acima, temos:

$$\Delta \Pi = - \left(\frac{\partial f}{\partial S} \mu S + \frac{\partial f}{\partial t} + \frac{1}{2} \frac{\partial^2 f}{\partial S^2} \sigma^2 S^2 \right) \Delta t - \frac{\partial f}{\partial S} \sigma S \Delta z + \frac{\partial f}{\partial S} (\mu S \Delta t + \sigma S \Delta z)$$

$$\Delta \Pi = - \left(\frac{\partial f}{\partial t} + \frac{1}{2} \frac{\partial^2 f}{\partial S^2} \sigma^2 S^2 \right) \Delta t$$

Como esta equação não envolve o termo Δz , a variação da carteira não tem risco durante um período curto de tempo Δt . Logo, esta carteira tem de possuir um retorno igual à taxa livre de risco r

$$\Delta \Pi = r \Pi \Delta t$$

Igualando os retornos das carteiras acima e adicionando o recebimento de uma taxa de dividendos constante q pela posse da ação S , a equação diferencial de Black Scholes é dada por :

$$-\left(\frac{\partial f}{\partial t} + \frac{1}{2} \frac{\partial^2 f}{\partial S^2} \sigma^2 S^2\right) \Delta t + \left(q \frac{\partial f}{\partial S}\right) \Delta t = r \left(-f + \frac{\partial f}{\partial S} S\right) \Delta t$$

$$rf = \frac{\partial f}{\partial S} + \frac{1}{2} \frac{\partial^2 f}{\partial S^2} \sigma^2 S^2 + (r - q) \frac{\partial f}{\partial S} S$$

A.4. Aproximação Analíticas

A.4.1. Barone-Adesi e Whaley

O valor de uma *call* americana é dada por:

$$V_{Call} = \begin{cases} V_{Call}^{BS}(S, X, T) + A_1 (S / S^*)^{q_1}, & S < S^* \\ S - X, & S \geq S^* \end{cases}$$

onde $A_1 = \frac{S^*}{q_1} (1 - e^{-q(T-t)} N(d_1(S^*)))$

$$d_1(S) = \frac{\ln(S/X) + (r - q + \sigma^2/2)(T-t)}{\sigma \sqrt{T-t}}$$

$$q_1 = \frac{-(2(r-q)/\sigma^2 - 1) + \sqrt{(2(r-q)/\sigma^2 - 1)^2 + \frac{8r}{\sigma^2(1 - e^{-r(T-t)})}}}{2}$$

onde S^* corresponde ao preço da ação cujo valor da *call* satisfaz a equação¹⁰ abaixo:

$$S^* - X = V_{Call}^{BS}(S^*, X, T) + \frac{1 - e^{-q(T-t)}}{q_1} N(d_1(S^*)) S^*$$

Para o caso de uma *put* americana, temos:

$$V_{Put} = \begin{cases} V_{Put}^{BS}(S, X, T) + A_2 (S/S^{**})^{q_2}, & S > S^{**} \\ S - X, & S \leq S^{**} \end{cases}$$

onde

$$A_2 = -\frac{S^{**}}{q_2} (1 - e^{-q(T-t)} N(-d_1(S^{**})))$$

$$q_2 = \frac{-\left(2(r-q)/\sigma^2 - 1\right) - \sqrt{\left(2(r-q)/\sigma^2 - 1\right)^2 + \frac{8r}{\sigma^2(1 - e^{-r(T-t)})}}}{2}$$

onde S^{**} corresponde ao preço da ação cujo valor da *put* satisfaz

$$S^{**} - X = V_{Put}^{BS}(S^{**}, X, T) + \frac{1 - e^{-q(T-t)}}{q_2} N(d_1(S^{**})) S^{**}$$

A.4.2. Bjerk Sund & Stensland

$$V_{Call} = \alpha S^\beta - \alpha \Phi(S, T-t, 1, I, I) + \Phi(S, T-t, 1, X, I) - X \Phi(S, T-t, 0, I, I) + X \Phi(S, T-t, 0, X, I)$$

para $\alpha = (I - X) I^{-\beta}$

$$b = r - q$$

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

¹⁰ Esta equação acima pode ser resolvida pelo método da bisseção ou Newton.

A função Φ é definida como:

$$\Phi(S, T, \gamma, H, I) = e^{\lambda} S^{\gamma} \left(N(d) - \left(\frac{I}{S} \right)^{\kappa} N\left(d - \frac{2 \ln(I/S)}{\sigma \sqrt{T}}\right) \right)$$

onde

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

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

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

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

$$B_0 = \max\left(X, \frac{r}{q} X\right)$$

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

$$I = B_0 + (B_{\infty} - B_0)(1 - e^{h(T)})$$

Para o caso de uma *put* americana, a aproximação do valor da opção pode ser feita por uma transformação na fórmula da *call* acima :

$$V_{put}(S, X, T, r, q, \sigma) = V_{call}(X, S, T, q, r, \sigma)$$

A.5. Aproximação de Geske e Johnson

Em seu artigo original Geske e Johnson mostram que uma opção americana pode ser estimada usando-se uma série de opções exercíveis em instantes finitos de tempo. A fórmula desenvolvida usa o método de Extrapolação de Richardson baseado numa série de opções do estilo bermuda. Assim, supondo que $P(n)$ seja o preço de uma *put* do estilo bermuda exercível em n instantes igualmente distantes, o valor de uma *put* americana pode ser aproximado por

$$P(1,2,3) = P(3) + \frac{7}{2}(P(3) - P(2)) - \frac{1}{2}(P(2) - P(1))$$

onde $P(1,2,3)$ representa o valor aproximado de uma *put* americana baseada em opções bermuda com uma, duas e três datas de exercício respectivamente.

Existem basicamente dois problemas nesta metodologia: (1) o modelo nem sempre converge pois existem casos onde $P(n) < P(m)$ para $m < n$; (2) existe uma certa dificuldade em determinarmos quantas opções bermuda ($P(1,2,3,\dots)$) devemos utilizar para atingirmos o nível de precisão desejada.

Uma segunda contribuição foi desenvolvida por Bunch e Johnson (1992) sugerindo uma modificação do modelo original de Geske e Johnson.

$$P(1,2) = P^{\max}(2) + (P^{\max}(2) - P(1))$$

onde $P^{\max}(2)$ corresponde ao valor da opção exercível em um de dois instantes de tempo, onde esses instantes são determinados de modo a maximizar o valor da opção. Em seu artigo Bunch e Johnson mostram que se os instantes de exercício forem determinados de forma a maximizarem $P(2)$, então o preço da *put* americana pode ser estimado de forma mais precisa que o modelo original de Geske e Johnson.

Outra sugestão proposta por Omberg, sugere outra modificação do modelo Geske e Johnson de modo a assegurar a convergência.

$$P(1,2,4) = P(4) + \frac{5}{3}(P(4) - P(2)) - \frac{1}{3}(P(2) - P(1))$$

onde novamente $P(1,2,4)$ representa o valor aproximado de uma *put* americana baseada em opções bermuda com uma, duas e quatro datas de exercício respectivamente. Assim, $P(4)$ corresponde ao valor da *put* exercível nos instantes $T/4$, $2T/4$, $3T/4$ e T . Por usarmos uma série geométrica, podemos assegurar que $P(4) \geq P(2) \geq P(1)$ sempre será válido. A razão para isto baseia-se no fato da

opção $P(4)$ incluir todos os instantes de exercício antecipado da opção $P(2)$, da mesma forma que $P(2)$ inclui o instante de exercício de $P(1)$.

A.6. Equações de Diferenças: Série de Taylor

Série de Taylor:

$$df = \frac{df}{dx} \cdot dx + \frac{1}{2} \cdot \frac{d^2 f}{dx^2} \cdot dx^2 + \frac{1}{6} \cdot \frac{d^3 f}{dx^3} \cdot dx^3 + \dots$$

$$f(x+dx) - f(x) = \frac{df}{dx} \cdot dx + \frac{1}{2} \cdot \frac{d^2 f}{dx^2} \cdot dx^2 + \frac{1}{6} \cdot \frac{d^3 f}{dx^3} \cdot dx^3 + \dots$$

$$f(x+dx) = f(x) + f'(x) \cdot dx + \frac{1}{2} \cdot f''(x) \cdot dx^2 + \frac{1}{6} \cdot f'''(x) \cdot dx^3 + \dots$$

Com base na escolha da diferença dx teremos diferentes tipos de aproximações para as derivadas parciais: *forward*, *backward* e *central*.

- *Derivadas de 1ª ordem:*

$$\text{Forward } (dx = h): f'(x) = \frac{f(x+h) - f(x)}{h} + \frac{1}{2} \cdot h^2 \cdot f''(x) + \frac{1}{6} \cdot h^3 \cdot f'''(x) + \dots$$

$$f'(x) = \frac{f(x+h) - f(x)}{h} + \varepsilon(h)$$

$$\text{Backward } (dx = -h): f'(x) = \frac{f(x) - f(x-h)}{h} + \varepsilon(h)$$

$$\text{Central } (dx = 2h): f'(x) = \frac{f(x+2h) - f(x)}{2h} + \varepsilon(h^2)$$

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + \varepsilon(h^2)$$

- *Derivada de 2º ordem:*

Resolver o sistema:

$$\begin{cases} f(x+h) = f(x) + f'(x) \cdot h + \frac{1}{2} \cdot f''(x) \cdot h^2 + \frac{1}{6} \cdot f'''(x) \cdot h^3 + \dots \\ f(x-h) = f(x) + f'(x) \cdot (-h) + \frac{1}{2} \cdot f''(x) \cdot (-h)^2 + \frac{1}{6} \cdot f'''(x) \cdot (-h)^3 + \dots \end{cases}$$

$$f(x+h) + f(x-h) = 2f(x) + f''(x) \cdot h^2 + \varepsilon(h^4)$$

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + \varepsilon(h^2)$$

A.7. Métodos Iterativos

Os métodos iterativos atingem a solução do sistema através do melhoramento de uma solução inicial a cada nova iteração. Dentre os inúmeros métodos optamos por estudar apenas os chamados métodos estacionários, que utilizam parâmetros que permanecem fixos durante as iterações. São exemplos deste modelo: métodos de Jacobi, Gauss-Seidel, SOR e SSOR.

- *Jacobi:*

Considere o sistema linear de equações,

$$\sum_{j=1}^N a_{ij} u_j = f_i, \quad i = 1, \dots, N$$

Se resolvêssemos o sistema para uma variável desconhecida, assumindo o conhecimento do valor das demais, teríamos a seguinte expressão:

$$u_i = \frac{1}{a_{ii}} \left(f_i - \sum_{j \neq i} a_{ij} u_j \right)$$

A equação acima sugere o seguinte algoritmo iterativo:

$$u_i^{n+1} = \frac{1}{a_{ii}} \left(f_i - \sum_{j \neq i} a_{ij} u_j^n \right),$$

onde n corresponde à iteração.

Dado a representação matricial do sistema de equações $A u = f$,

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{NN} \end{pmatrix}, \quad \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_N \end{bmatrix} \quad \text{e} \quad \mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{bmatrix}$$

e decompondo a matriz A , temos:

$$\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U},$$

onde:

$$\mathbf{D} = \begin{pmatrix} a_{11} & & & & \\ & a_{22} & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & a_{NN} \end{pmatrix}, \quad \mathbf{L} = - \begin{pmatrix} 0 & & & & \\ a_{21} & 0 & & & \\ \vdots & \vdots & \ddots & & \\ a_{N1} & a_{N2} & \cdots & 0 \end{pmatrix}$$

$$\text{e} \quad \mathbf{U} = - \begin{pmatrix} 0 & a_{12} & \cdots & a_{1N} \\ & 0 & \cdots & a_{2N} \\ & & \ddots & \vdots \\ & & & 0 \end{pmatrix}$$

Em notação vetorial, o método de Jacobi é dado por:

$$\mathbf{u}^{n+1} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})\mathbf{u}^n + \mathbf{D}^{-1}\mathbf{f}$$

onde \mathbf{u}^{n+1} representa o vetor de variáveis desconhecidas u_i^{n+1} , $i = 1, \dots, N$. Assim \mathbf{D} , $-\mathbf{L}$ e $-\mathbf{U}$ representam as matrizes diagonal, triangular inferior e superior da respectiva matriz \mathbf{A} .

- *Gauss-Seidel*:

Corresponde a uma generalização do método de Jacobi. A única diferença é que as alterações nas variáveis são incorporadas ao processo à medida que ocorrem. O algoritmo do método é dado por:

$$u_i^{n+1} = \frac{1}{a_{ii}} \left(f_i - \sum_{j<i} a_{ij} u_j^{n+1} - \sum_{j>i} a_{ij} u_j^n \right)$$

Em notação vetorial teríamos:

$$\mathbf{u}^{n+1} = (\mathbf{D} - \mathbf{L})^{-1} (\mathbf{U} \mathbf{u}^n + \mathbf{f})$$

- *Successive overrelaxation method (SOR)*:

O método SOR é constituído pela ponderação de duas iterações sucessivas do Gauss-Seidel.

$$\tilde{u}_i^{n+1} = \frac{1}{a_{ii}} \left(f_i - \sum_{j<i} a_{ij} u_j^{n+1} - \sum_{j>i} a_{ij} u_j^n \right)$$

$$u_i^{n+1} = \omega \tilde{u}_i^{n+1} + (1 - \omega) u_i^n$$

O parâmetro ω é conhecido como *overrelaxation parameter*. O seu valor afeta enormemente a taxa de convergência, sendo seu valor ótimo de difícil determinação.

Se optarmos por $\omega = 1$, o método SOR transforma-se no Gauss-Seidel. Estudos tem demonstrado que o método não converge para valores de ω fora do intervalo $[0, 2]$.

Em notação vetorial teríamos:

$$v^{n+1} = (\mathbf{D} - \omega \mathbf{L})^{-1} (\omega \mathbf{U} + (1 - \omega) \mathbf{D}) u^n + \omega (\mathbf{D} - \omega \mathbf{L})^{-1} \mathbf{f}$$

- *Symmetric Successive overrelaxation method (SSOR)*:

Em notação vetorial, o algoritmo SSOR é dado por:

$$u^{n+1} = \mathbf{B}_1 \mathbf{B}_2 u^n + \omega (2 - \omega) (\mathbf{D} - \omega \mathbf{U})^{-1} \mathbf{D} (\mathbf{D} - \omega \mathbf{L})^{-1} \mathbf{f}$$

onde $\mathbf{B}_1 = (\mathbf{D} - \omega \mathbf{U})^{-1} (\omega \mathbf{L} + (1 - \omega) \mathbf{D})$;

$$\mathbf{B}_2 = (\mathbf{D} - \omega \mathbf{L})^{-1} (\omega \mathbf{U} + (1 - \omega) \mathbf{D}).$$

A.8. Método da Bisseção

Seja $f(x)$ uma função contínua definida no intervalo $I_0 = [a, b]$ tal que $f(a).f(b) < 0$. Para simplificar, suponha também que nesse intervalo exista uma única raiz. Em cada iteração, a amplitude do novo intervalo I_i será reduzida pela

metade. Assim temos que: $x = \frac{a+b}{2}$

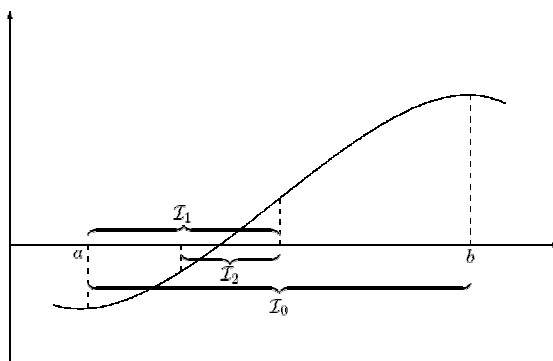


Ilustração A.1- Esquema do método da bisseção.

O novo intervalo é obtido da seguinte maneira: se $f(x).f(b) < 0$ então o novo intervalo é $I_1 = [a, x]$, caso contrário, $f(x).f(b) > 0$, $I_1 = [x, b]$. Repetimos

esse processo até que o tamanho do intervalo seja suficientemente pequeno (ϵ). A raiz pode então ser aproximada por qualquer número pertencente ao intervalo.

A.9. Normalização de Seqüências $U \sim (0,1)$

- *Box-Muller:*

Através de duas seqüências $U \sim (0,1)$ independentes, usaremos o método de Box-Muller para transformá-las em distribuições $N \sim (0,1)$. Assim, temos:

Resumindo: $(u_1, u_2) \sim U(0,1) \xrightarrow{\text{Box-Muller}} (x, y) \sim N(0,1)$

$$x = \sqrt{-2 \log(u)} \text{sen}(2\pi v)$$

$$y = \sqrt{-2 \log(u)} \cos(2\pi v)$$

Conforme observamos, para utilizarmos o Método de Box-Muller devemos gerar no mínimo duas seqüências de variáveis quase-aleatórias.

- *Inversão de Moro:*

O algoritmo desenvolvido por Moro divide o domínio y em duas regiões:

- 1- a região central da distribuição, $|y| \leq 0.42$, é calculada com base na aproximação de Beasley e Springer (1977);
- 2- as caudas da distribuição, $|y| > 0.42$, são modeladas com base nas séries de Chebyshev.

A função distribuição normal acumulada é dada por:

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt$$

Assim, dado $y = x - 0.5$ ($x \sim U(0,1)$), temos:

$$\text{Para } y \leq 0.42 : \quad \Phi^{-1}(x) = y \frac{\sum_{n=0}^3 a_n y^{2n}}{\sum_{n=0}^4 b_n y^{2n}}$$

$$\text{Para } y > 0.42 : \quad \Phi^{-1}(x) = \begin{cases} \sum_{n=0}^8 c_n T_n(z) - \frac{c_0}{2}, & y > 0 \\ \frac{c_0}{2} - \sum_{n=0}^8 c_n T_n(z), & y \leq 0 \end{cases}$$

$$\therefore \quad z = k_1 [2. \log(-\log(0.5 - |y|)) - k_2]$$

onde k_1 e k_2 são escolhidos de modo a termos $z = -1$ quando $\Phi(x) = 0.92$ e $z = 1$ quando $\Phi(x) = 1 - 10^{-12}$. As constantes, a_n, b_n, c_n, k_1 e k_2 são dadas pela tabela abaixo:

Tabela A.1- Parâmetros¹¹ do método de inversão de Moro.

n	a_n	b_n	n	c_n
0	2.50662823884	1.00	0	7.7108870705487895
1	-18.61500062529	-8.47351093090	1	2.7772013533685169
2	41.39119773534	23.08336743743	2	0.3614964129261002
3	-25.44106049637	-21.06224101826	3	0.0373418233434554
4		3.13082909833	4	0.0028297143036967
	k_1	k_2	5	0.0001625716917922
	0.4179886424926431	4.2454686881376569	6	0.0000080173304740
			7	0.0000003840919865
			8	0.0000000129707170

A.10. Discrepância

Uma medida comum de homogeneidade baseia-se na idéia de como um conjunto de d vetores de números estão dispersos num cubo multidimensional unitário. A interpretação geométrica pode ser melhor definida a seguir: Devemos gerar d seqüências quase-aleatórias uniformes $(\{r_i\}^d, i = 1, \dots, N)$, compostas por N números. Esses vetores de números podem ser vistos como as coordenadas de

¹¹ Valores sugeridos por Joy, Boyle e Tan.

pontos do cubo unitário de dimensão igual a d ($[0,1]^d$). Agora, selecionaremos d números compreendidos entre 0 e 1 de modo a gerarmos um retângulo d dimensional representado por uma sub-região $S(y) = [(0, y_1) \times (0, y_2) \times \dots \times (0, y_d)]$. Posteriormente, definiremos $n_{S(y)}$ como o total de números compreendidos dentro desta sub-região.

A medida que $N \rightarrow \infty$, observamos que para um gerador de números quase-aleatórios possuir uma homogeneidade perfeita temos que:

$$\lim_{N \rightarrow \infty} \frac{n_{S(y)}}{N} = \prod_{i=1}^d y_i, \text{ para todo } y \in [0,1]^d$$

A equação acima resulta do fato de que para uma distribuição uniforme perfeitamente homogênea, a probabilidade de um número da seqüência estar dentro da sub-região $S(y)$ deve ser igual ao volume da própria sub-região, definido como $V_{S(y)} = \prod_{i=1}^d y_i$. Com essa definição, podemos comparar $n_{S(y)}/N$ com $V_{S(y)}$ para cada uma das seqüências de forma a obtermos uma medida de erro para sua discrepância geral, definida como:

$$T_N^{(d)} = \left(\int_{[0,1]^d} \left(\frac{n_{S(y)}}{N} - \prod_{k=1}^d y_k \right)^2 dy \right)^{\frac{1}{2}}.$$

A.11. Método dos Mínimos Quadrados

$$\begin{aligned} Y_i &= \hat{\alpha}_1 + \hat{\alpha}_2 X_i + e_i & \therefore & E[u_i] = 0, \\ & & & Var[u_i] = \sigma^2, \\ & & & Cov[u_i, u_j] = 0, \text{ para } \forall i \end{aligned}$$

A equação acima estabelece para cada observação i uma relação linear de dependência – suportada por dois parâmetros estimados $\hat{\alpha}_1$ e $\hat{\alpha}_2$, denominados

coeficientes de regressão – entre as variáveis observadas Y e X e um resíduo de estimação para a i -ésima observação e_i , onde:

$$e_i = Y_i - \hat{Y}_i \quad \therefore \quad \hat{Y}_i = \hat{\alpha}_1 + \hat{\alpha}_2 X_i$$

Resulta das hipóteses acima: $E[Y_i] = \hat{\alpha}_1 + \hat{\alpha}_2 X_i$ e $Var[Y_i] = \sigma^2$.

Na estimação dos valores α_1 e α_2 , adotamos o método dos mínimos quadrados (MQ), que se baseia no critério de minimização da soma dos quadrados dos resíduos:

$$\sum_{i=1}^n e_i^2 = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 = \sum_{i=1}^n (Y_i - \hat{\alpha}_1 - \hat{\alpha}_2 X_i)^2 \quad \therefore$$

onde n – representa o número de observações.

Os estimadores são então definidos como:

$$\hat{\alpha}_1 = \bar{Y} - \hat{\alpha}_2 \bar{X} \quad \text{e} \quad \hat{\alpha}_2 = \frac{n \sum_{i=1}^n X_i Y_i - \sum_{i=1}^n X_i \sum_{i=1}^n Y_i}{n \sum_{i=1}^n X_i^2 - \left(\sum_{i=1}^n X_i \right)^2}$$

ou

$$\hat{\alpha}_2 = \frac{\sum_{i=1}^n x_i y_i}{\sum_{i=1}^n x_i^2} \quad \text{onde} \quad y_i = Y_i - \bar{Y}, \quad x_i = X_i - \bar{X}$$

A.12. Polinômios

- *Legendre*

$$y(x) = \sum_{k=0}^n \alpha_k P_k(x)$$

para $P_0(x) = 1, P_1(x) = x, P_2(x) = \frac{1}{2}(3x^2 - 1)$

.....

$$P_{n+1}(x) = \frac{2n+1}{n+1}xP_n(x) - \frac{n}{n+1}P_{n-1}(x)$$

• *Laguerre:*

$$y(x) = \sum_{k=0}^n \alpha_k L_k(x)$$

para $L_0(x) = 1, L_1(x) = 1 - x, L_2(x) = 2 - 4x + x^2,$

.....

$$L_{n+1}(x) = (1 + 2n - x)L_n(x) - n^2 L_{n-1}(x)$$

• *Hermite:*

$$y(x) = \sum_{k=0}^n \alpha_k H_k(x)$$

para $H_0(x) = 1, H_1(x) = 2x, H_2(x) = 4x^2 - 2,$

.....

$$H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x)$$

A.13. Fatoração LU

Dado a representação matricial do sistema de equações $\mathbf{A}\mathbf{x} = \mathbf{f}$,

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{NN} \end{pmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{bmatrix} \quad \text{e} \quad \mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_N \end{bmatrix}$$

Podemos decompor a matriz **A** de forma que:

$$\mathbf{P.A} = \mathbf{L.U}$$

tal que, **P** é uma matriz de pivoteamento cujos valores são 0 ou 1, e **L** e **U** são matrizes triangulares inferior e superior respectivamente. Assim, a solução do sistema (**x**) pode ser obtida pela solução trivial de dois sistemas:

$$\mathbf{L.y} = \mathbf{P.f} \quad \text{e} \quad \mathbf{U.x} = \mathbf{y}$$

A.14. Números de Inicialização de Sobol

Tabela A.2- Números de inicialização da seqüência de Sobol com seus respectivos polinômios primitivos até a dimensão 32.

N	G _k	a ₀a _k	Números de inicialização (m ⁰ m ¹⁰)										
1	0		1	1	1	1	1	1	1	1	1	1	1
2	1	11	1	3	5	15	17	51	85	255	257	771	
3	2	111	1	1	7	11	13	61	67	79	465	721	
4	3	1011	1	3	7	5	7	43	49	147	439	1013	
5	3	1101	1	1	5	3	15	51	125	141	177	759	
6	4	10011	1	3	1	1	9	59	25	89	321	835	
7	4	11001	1	1	3	7	31	47	109	173	181	949	
8	5	100101	1	3	3	9	9	57	43	43	225	113	
9	5	101001	1	3	7	7	21	61	55	19	59	761	
10	5	101111	1	1	5	11	27	53	69	25	103	615	
11	5	110111	1	1	5	3	29	51	47	97	233	39	
12	5	111011	1	3	7	13	3	35	89	9	235	929	
13	5	111101	1	3	5	1	15	19	113	115	411	157	
14	6	1000011	1	1	1	9	23	37	97	97	353	169	
15	6	1011011	1	1	3	13	11	7	37	101	463	657	
16	6	1100001	1	3	3	5	19	33	3	197	329	983	
17	6	1100111	1	1	7	13	25	5	27	71	377	719	
18	6	1101101	1	1	1	3	13	39	7	23	391	389	
19	6	1110011	1	3	5	11	7	11	43	25	187	825	
20	7	10000011	1	3	1	7	3	23	79	65	451	321	
21	7	10001001	1	3	1	15	17	63	13	113	147	881	
22	7	10001111	1	3	3	3	25	17	115	17	179	883	
23	7	10010001	1	3	7	9	31	29	17	121	363	783	
24	7	10011101	1	1	3	15	29	15	41	249	201	923	
25	7	10100111	1	3	1	9	5	21	119	53	319	693	
26	7	10101011	1	1	5	5	1	27	33	253	341	385	
27	7	10111001	1	1	3	1	23	13	75	29	181	895	
28	7	10111111	1	1	7	7	19	25	105	173	509	75	
29	7	11000001	1	3	5	5	21	9	7	143	157	959	
30	7	11001011	1	1	1	15	5	49	59	71	31	111	
31	7	11010011	1	3	5	15	17	19	21	227	413	727	
32	7	11010101	1	1	7	11	13	29	3	15	279	17	

Apêndice B:

PROGRAMAS

No estudo dos modelos numéricos abordados optei por desenvolver os programas em MatLab por ser esse um *software* bastante utilizado em finanças. Assim, apesar do tempo adicional gasto para implementar os modelos, tive domínio total das variáveis de entrada e saída dos programas. Isso significa que tive maior flexibilidade para explorar cada modelo e apresentar resultados diversos que não apenas a simples precificação de opções.

Abaixo, apresentamos a lista de programas resultantes deste trabalho assim como, as interfaces computacionais de três programas desenvolvidos com intuito de afirmar a possibilidade de utilizarmos o *software* e modelos estudados no desenvolvimento de aplicativos financeiros.

B.1. Lista de Programas

NOME	DESCRIÇÃO
<i>American_GVW</i>	Avaliação de opções americanas pelo Modelo de GVW
<i>American_LSM</i>	Avaliação de opções americanas pelo Modelo LSM
<i>American_DF</i>	Avaliação de uma put americana pelo método de DF
<i>American_JumpRuin</i>	Avaliação de opções americanas utilizando o processo estocástico jump-to-ruin
<i>American_DownOutPut</i>	Avaliação de uma put americana do tipo barreira <i>down-out</i>
<i>American_UpOutPut</i>	Avaliação de uma put americana do tipo barreira <i>up-out</i>
<i>American_Asian</i>	Avaliação de opções americanas asiáticas (média aritmétrica ou geométrica)
<i>American_LookbackS</i>	Avaliação de opções americanas <i>lookback floating strike</i> (valor de exercício flutuante)
<i>American_LookbackX</i>	Avaliação de opções americanas <i>lookback floating price</i> (preço da ação flutuante)
<i>American_Best2</i>	Avaliação de opções americanas compostas por dois ativos correlacionados
<i>American_Best3</i>	Avaliação de opções americanas compostas por três ativos correlacionados
<i>American_Juros</i>	Avaliação de opções americanas com taxas de juros variáveis
<i>American_TaxaJuros_correl</i>	Avaliação de opções americanas com taxas de juros estocásticas (CIR) e correlacionadas com o ativo
<i>American_NGARCH</i>	Avaliação de opções americanas com volatilidade estocástica (NGARCH)
<i>AmerCall_Barone</i>	Precificação de uma call americana pela aproximação analítica de Barone-Adesi & Whaley
<i>American_BjSt</i>	Precificação de uma call/put americana pela aproximação analítica de de Bjerksund & Stensland
<i>MudBase</i>	Mudança da base decimal de um número inteiro 'N' para a base 'b'
<i>Halton</i>	Gera um número quase-aleatório com base 'b' em Halton
<i>SeqHaltonBase</i>	Gera uma sequência com 'n' números quase-aleatórios de Halton
<i>SeqHaltonDim</i>	Gera uma sequência de baixa discrepância de Halton com dimensão 'D' (<100)
<i>SeqFaure</i>	Gera uma sequência de baixa discrepância de Faure
<i>DiracSobol</i>	Gera os números direcionais da sequência de Sobol.
<i>SeqSobol</i>	Gera uma sequência de baixa discrepância de Sobol com base nos números direcionais
<i>SeqSobolDim</i>	Gera uma sequência de Sobol de dimensão 'D' (<100)
<i>VetRandPerm</i>	Gera sequências baseadas no modelo de QMC Híbrido
<i>Moro</i>	Transforma uma sequência Uniforme (0,1) em Normal (0,1) com base no método de inversão de Moro
<i>Box_Muller</i>	Transforma duas sequências Uniformes (0,1) em uma Normal (0,1) com base no método de Box-Muller
<i>EurCall_MC</i>	Precificação de uma call europeia por SMC
<i>EurCall_Halton</i>	Precificação de uma call europeia por QMC usando números aleatórios de Halton
<i>EurCall_Hibrido</i>	Precificação de uma call europeia por QMC-Híbrido (Halton)
<i>EurCall_Faure</i>	Precificação de uma call europeia por QMC usando números aleatórios de Faure
<i>EurCall_Sobol</i>	Precificação de uma call europeia por QMC usando números aleatórios de Sobol
<i>EuroCall_AV</i>	Precificação de uma call europeia por MC usando o método das variáveis antitéticas
<i>EuroCall_CV</i>	Precificação de uma call europeia por MC usando variáveis de controle
<i>EuroCall_SS</i>	Precificação de uma call europeia por MC usando estratificação
<i>EuroCall_IS</i>	Precificação de uma call europeia por MC usando <i>importance sampling</i>
<i>Integral_MC</i>	Cálculo de uma integral (simples ou dupla) por SMC
<i>Integral_Halton</i>	Cálculo de uma integral (simples ou dupla) por QMC-Halton
<i>Integral_Faure</i>	Cálculo de uma integral (simples ou dupla) por QMC-Faure
<i>Integral_Sobol</i>	Cálculo de uma integral (simples ou dupla) por QMC-Sobol
<i>Integral_AV</i>	Cálculo de uma integral simples por SMC usando o método de variáveis antitéticas
<i>Integral_CV</i>	Cálculo de uma integral simples por SMC usando o método de variáveis de controle
<i>Integral_SS</i>	Cálculo de uma integral simples por SMC usando o método de estratificação
<i>AmericanPut_Crank_SOR</i>	Precificação de uma put americana por DF Crank-Nicholson (SOR)
<i>Americanput_expl</i>	Precificação de uma put americana por DF Explícito
<i>Americanput_expl_PROB</i>	Gráfico as probabilidades do método de DF Explícito
<i>AmericanPut_Impl</i>	Precificação de uma put americana por DF Implícito (Brennan & Schwartz)
<i>AmericanPut_Impl_SOR</i>	Precificação de uma put americana por DF Implícito (SOR)
<i>AmericanPut_Impl_SSOR</i>	Precificação de uma put americana por DF Implícito (SSOR)
<i>SOR_American</i>	Solução de um sistema pelo método SOR e programação dinâmica (usado na função <i>AmericanPut_Impl_SOR</i>)
<i>SSOR_American</i>	Solução de um sistema pelo método SSOR e programação dinâmica (usado na função <i>SSOR_American</i>)
<i>Regressão</i>	Regressão dos mínimos quadrados usando polinômios de Legendre, Hermite, Lagrange e linear
<i>BinAm_BBSR</i>	Precificação de uma call/put europeia pelo modelo binomial BBS c/ Extrapolacao de Richardson
<i>BinAm_BS</i>	Precificação de uma call/put europeia pelo modelo binomial Black-Scholes
<i>BinAm_Control</i>	Precificação de uma call/put americana pelo modelo binomial c/ variável de controle
<i>BinAm_MVM</i>	Precificação de uma call/put americana pelo modelo binomial c/ valores médios
<i>BinAmCall</i>	Precificação de uma call americana pelo modelo binomial
<i>BinAmCall_BS</i>	Precificação de uma call americana pelo modelo binomial Black-Scholes
<i>BinAmPut</i>	Precificação de uma put americana pelo modelo binomial

Quadro B.1- Lista de programas desenvolvidos.

B.2. Interface Computacional

B.2.1. Modelo de Diferenças Finitas

Calculadora DF:

PUT Americana: Modelo de Diferenças Finitas

Características da opção:

Preço inicial (S0): 60 Strike price (K): 60 Tempo (T): 6/12 Taxa sem risco (r): 0.1 Dividendos (q): 0.02 DP (sigma): 0.35

Opcoes:

GRID: Ex GRID Metodo: Graficos

M: 100 N: 100 S max: 100 **Implicito SOR**

SOR

Omega (w): .4
Interacoes: 100
Tolerancia: .00001

RESULTADOS:

Aprox. BJS:	PUT	Tempo	Erro (%)
4.83628	4.8812	00:00:03	0.9286

Gregas:

Delta	Gama	Theta
-0.41409	0.029176	-3.9575

Calcular

Figura B.1- Interface computacional do programa *American_DF.m*.

B.2.2. Modelo GVW

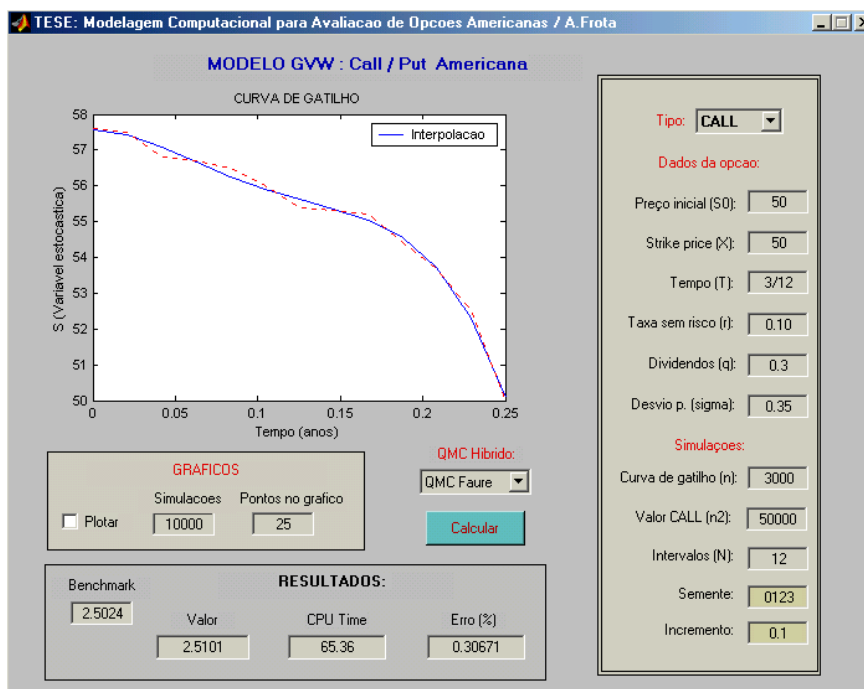


Figura B.2- Interface computacional do programa *American_GVW.m*.

B.2.3. Modelo LSM

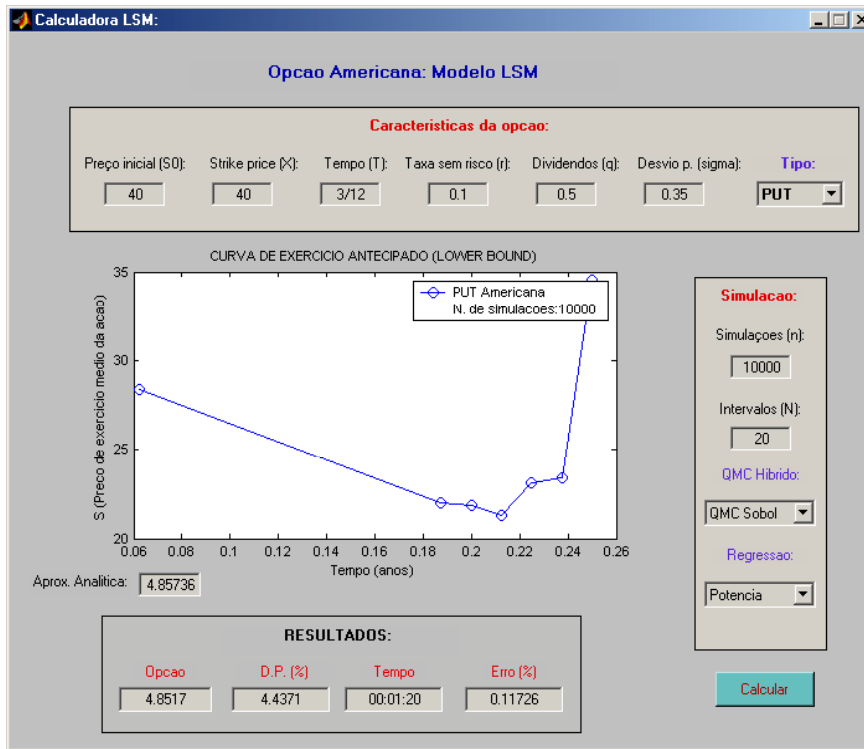


Figura B.3- Interface computacional do programa *American_LSM.m*.