

FUNDAÇÃO GETÚLIO VARGAS
ESCOLA DE ECONOMIA DE SÃO PAULO

GUIDO MARCELO BORMA CHAGAS

**PRECIFICAÇÃO DE OPÇÕES DE DÓLAR NO MERCADO BRASILEIRO
UTILIZANDO REDES NEURAIS E ALGORITMOS GENÉTICOS**

SÃO PAULO
2006

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

GUIDO MARCELO BORMA CHAGAS

**PRECIFICAÇÃO DE OPÇÕES DE DÓLAR NO MERCADO BRASILEIRO
UTILIZANDO REDES NEURAIS E ALGORITMOS GENÉTICOS**

Dissertação apresentada à Escola de Economia de São Paulo da Fundação Getúlio Vargas como requisito para obtenção do título de Mestre em Finanças e Economia Empresarial.

Campo de Conhecimento:
Finanças, Derivativos

Orientador: Prof. Dr. Ricardo Ratner
Rochman

SÃO PAULO
2006

Chagas, Guido Marcelo Borma
Precificação de Opções de Dólar no Mercado Brasileiro
utilizando Redes Neurais e Algoritmos Genéticos / Guido
Marcelo Borma Chagas.
88f.

Orientador: Prof. Dr. Ricardo Ratner Rochman.
Dissertação (mestrado) – Fundação Getúlio Vargas.
Escola de Economia de São Paulo. São Paulo, 2006.
Área de concentração: Finanças

1. Finanças 2. Opções 3. Inteligência Artificial

GUIDO MARCELO BORMA CHAGAS

**PRECIFICAÇÃO DE OPÇÕES DE DÓLAR NO MERCADO BRASILEIRO
UTILIZANDO REDES NEURAIS E ALGORITMOS GENÉTICOS**

Dissertação apresentada à Escola de Economia de São Paulo da Fundação Getúlio Vargas como requisito para obtenção do título de Mestre em Finanças e Economia Empresarial.

Campo de Conhecimento:
Finanças, Derivativos

Data de Aprovação:

___/___/___

Banca Examinadora:

Prof. Dr. Ricardo Ratner Rochman (Orientador)
FGV-EESP

Prof. Dr. Afonso de Campos Pinto
FGV-EESP

Prof. Dr. Elmo Tambosi Filho
UFSC-PPGEP

SÃO PAULO
2006

AGRADECIMENTOS

Aos meus pais Vera e Wilson cuja dedicação, amor e orientação possibilitaram que eu alcançasse meus objetivos, ao meu irmão Thiago e à minha tia Lia, cujos conselhos bem humorados me ajudaram a superar os momentos difíceis e à minha amiga e companheira Anelise, cuja paciência, apoio e amor foram imprescindíveis para que eu concluísse com sucesso essa etapa da minha vida.

Ao prof. Dr. Ricardo cuja experiência e paciência me auxiliaram a manter o foco e a concentração dos meus esforços nos pontos de maior relevância para este trabalho.

Aos meus grandes amigos Daniel, Eliza, Luciana, Márcio e Rodrigo, cuja companhia, motivação, persistência e amizade estiveram sempre presentes nestes últimos anos.

E por fim, aos profissionais do BankBoston e do Itaú pela compreensão e suporte oferecidos e a todos aqueles que, direta ou indiretamente, contribuíram com a elaboração deste trabalho.

Aos meus pais Vera e Wilson.

RESUMO

Esse trabalho comparou, para condições macroeconômicas usuais, a eficiência do modelo de Redes Neurais Artificiais (RNAs) otimizadas por Algoritmos Genéticos (AGs) na precificação de opções de Dólar à Vista aos seguintes modelos de precificação convencionais: Black-Scholes, Garman-Kohlhagen, Árvores Trinomiais e Simulações de Monte Carlo.

As informações utilizadas nesta análise, compreendidas entre janeiro de 1999 e novembro de 2006, foram disponibilizadas pela Bolsa de Mercadorias e Futuros (BM&F) e pelo Federal Reserve americano.

As comparações e avaliações foram realizadas com o software MATLAB, versão 7.0, e suas respectivas caixas de ferramentas que ofereceram o ambiente e as ferramentas necessárias à implementação e customização dos modelos mencionados acima.

As análises do custo do delta-hedging para cada modelo indicaram que, apesar de mais complexa, a utilização dos Algoritmos Genéticos exclusivamente para otimização direta (binária) dos pesos sinápticos das Redes Neurais não produziu resultados significativamente superiores aos modelos convencionais.

Palavras-chave: derivativos, opções, algoritmos genéticos, redes neurais.

ABSTRACT

This work compared, under usual macroeconomic conditions, the effectiveness of the Neural Networks (NN) model enhanced by Genetic Algorithms (GA) in Dollar options' valuation with the following conventional valuation models: Black-Scholes, Garman-Kohlhagen, Trinomial Trees and Monte Carlo Simulations.

All information employed in this analysis, comprehended between July, 1999 and December, 2006, was provided by Bolsa de Mercadorias e Futuros (BM&F) and by Federal Reserve.

Comparisons and assessments were conducted with the MATLAB software, version 7.0, and its toolboxes which provided the necessary tools and environment to develop and implement the models previously mentioned.

The delta-hedging cost's analyses of each model indicated that, even though more complex, the use of Genetic Algorithms to directly optimize (i.e., at binary level) the Neural Network's synaptic weights did not produce any significantly superior results than the conventional models.

Keywords: derivatives, options, genetic algorithms, neural networks.

ÍNDICE DE FIGURAS

Figura 2.2.2.1: Comportamento Binomial do Portifólio $\Delta \cdot S - C$ para um Único Período	9
Figura 2.2.2.2: Comportamento Binomial do Portifólio $\Delta \cdot S - C$ para N Períodos	13
Figura 2.2.2.3: Comportamento Trinomial do Portifólio $\Delta \cdot S - C$ para Um Único Período	14
Figura 2.2.4.1.1: Modelo de um Neurônio Não Linear	20
Figura 2.2.4.1.2: Exemplos de Arquiteturas de RNAs	20
Figura 2.2.4.1.3: Fluxo de Informações de um Neurônio da Camada de Saída	24
Figura 2.2.4.1.4: Fluxo de Informações de um Neurônio da Camada Intermediária (Invisível)	27
Figura 2.2.4.2.1: Comportamentos dos Diferentes Tipos de Cruzamento	36
Figura 2.2.4.2.2: Comportamentos do Processo de Mutação	37
Figura 3.1: Funções de Ativação Utilizadas	46
Figura 3.2: Topologia da RNA com Configuração 10-5-1	47
Figura 3.3: Topologia da RNA com Configuração 5-2-1 com Recorrência	48
Figura 3.4: Topologia da RNA com Configuração 12-4-1	50
Figura 3.5: Topologia da RNA com Configuração 7-4-1	51
Figura 3.6: Topologias das RNAs com Configurações 7-1 e 5-1 com Recorrência	52
Figura 3.7: Topologia com Melhor Desempenho	53
Figura 4.1: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 10-5-1	58
Figura 4.2: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 6-3-1	58
Figura 4.3: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 5-2-1 com Recorrência	59
Figura 4.4: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 12-4-1	59
Figura 4.5: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 7-1	60
Figura 4.6: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 5-2-1	60
Figura 4.7: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 5-2-1 sem Regularização Bayesiana	61
Figura 4.8: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com configuração 5-2-1	63

ÍNDICE DE TABELAS

Tabela 3.1: Resumo das Configurações de RNA com AG Analisadas	45
Tabela 4.1: Comparação dos Custos de Delta-Hedging dos Modelos Analisados	66

SUMÁRIO

1	INTRODUÇÃO	1
2	REVISÃO BIBLIOGRÁFICA	3
2.1	Redes Neurais e Algoritmos Genéticos Aplicados a Precificação de Derivativos	3
2.2	Modelos de Precificação	4
2.2.1	Modelos de Black-Scholes, Black e Garman-Kohlhagen	4
2.2.2	Árvores Binomiais e Trimoniais	8
2.2.3	Simulações de Monte Carlo	16
2.2.4	Redes Neurais Artificiais otimizadas por Algoritmos Genéticos	19
2.2.4.1	Redes Neurais Artificiais	19
2.2.4.2	Algoritmos Genéticos	32
2.2.4.3	Combinando Redes Neurais Artificiais e Algoritmos Genéticos	38
3	METODOLOGIA	42
4	ANÁLISES	56
5	CONSIDERAÇÕES FINAIS	68
6	REFERÊNCIAS BIBLIOGRÁFICAS	70

1 INTRODUÇÃO

A utilização de modelos adequados de precificação é um aspecto fundamental para o sucesso de diversas estratégias utilizadas no mercado financeiro. Segundo McNelis (2005), a simplicidade, a velocidade e a precisão são parâmetros essenciais à avaliação desta adequação.

O modelo de Black (1976), por exemplo, amplamente utilizado pelo mercado doméstico segundo Lanari, Souza e Duque (1999) e com uma forma analítica fechada, oferece menor complexidade e tempo de resposta do que o modelo de Simulações de Monte Carlos.

Sua precisão, entretanto, é inferior àquelas de outros modelos mais elaborados como, por exemplo, Black, Derman e Toy (1990), desenvolvidos especialmente para captar determinadas assimetrias e peculiaridades das distribuições financeiras convencionais.

Segundo a abordagem de McNelis (2005), podemos considerar a simplicidade e a velocidade como restrições binárias onde, atendidos um nível máximo de complexidade (que viabilize sua implementação) e um tempo máximo de resposta (que permita sua utilização prática), quaisquer melhorias não são relevantes ao resultado da estratégia.

A precisão, entretanto, impacta diretamente na avaliação do risco e retorno da estratégia influenciando nas decisões de manutenção, aumento ou redução da exposição nos ativos e instrumentos derivativos.

Por esta razão, o desenvolvimento de modelos mais sofisticados e robustos, baseados em ferramentas e conceitos mais complexos, tem sido beneficiado pela evolução tecnológica. Modelos numéricos de precificação recentes tais como Black,

Derman e Toy (1990) e Heath, Jarrow e Morton (1992) são exemplos de aplicações práticas altamente dependentes da tecnologia disponível.

O objetivo deste trabalho foi comparar a precisão de um modelo conceitualmente mais complexo: o modelo de Redes Neurais Artificiais otimizadas por Algoritmos Genéticos à precisão na precificação de opções de Dólar à Vista dos modelos de Black-Scholes, Garman-Kohlhagen, Árvores Trinomiais e Simulações de Monte Carlo, em condições macroeconômicas usuais.

2 REVISÃO BIBLIOGRÁFICA

2.1 Redes Neurais e Algoritmos Genéticos Aplicados a Precificação de Derivativos

Hutchinson *et al* (1994) foi um dos primeiros estudos a utilizar Redes Neurais na precificação de opções. Em seu trabalho, envolvendo contratos futuros de índice S&P 500 negociados entre 1987 e 1991, os autores destacaram que os modelos de Redes Neurais oferecem maior precisão quando o ativo-base apresenta uma dinâmica de preços desconhecida ou quando a solução da equação diferencial do derivativo não pode ser obtida analiticamente.

Qi *et al* (1996) realizou análise semelhante para os contratos futuros de S&P 500 negociados entre 1994 e 1995 utilizando uma rede Multi-Layer Perceptron (MLP), significativamente mais eficiente que suas antecessoras melhorando os tempos de resposta destes modelos.

Como alternativa às metodologias mais tradicionais de precificação de opções utilizando Redes Neurais, Chen e Lee (1997) propuseram o uso de Algoritmos Genéticos. Em seu trabalho, eles enfatizaram a superioridade desta abordagem na análise de derivativos que apresentem superfícies de erro mais complexas.

Posteriormente, De Falco (1998) demonstrou que uma solução híbrida envolvendo Algoritmos Genéticos e Redes Neurais proporciona melhor performance e eficiência, particularmente nos casos propensos a mínimos locais. Quagliarella e Viani (1998) acrescentaram que a utilização destes modelos híbridos, compostos pela combinação de mecanismos de Back-Propagation com os Algoritmos Genéticos, oferecem a melhor relação desempenho / precisão.

No mercado financeiro brasileiro, um dos primeiros trabalhos nesta área foi elaborado por Fernandes (2000), que utilizou Redes Neurais na precificação e hedge dinâmico das opções de Telebrás. Em seu trabalho, ele apresentou algumas limitações desta metodologia para o mercado local, particularmente quando aplicada a precificação de ativos ilíquidos ou com histórico de preços reduzido.

Outros trabalhos interessantes no Brasil foram desenvolvidos por Freitas (2001), que sugeriu que a precificação de opções de Telebrás por Redes Neurais proporciona resultados superiores aos do modelo de Black-Scholes convencional, e por Kamakura (2004), que utilizou Algoritmos Genéticos para aperfeiçoar mecanismos de hedge paramétricos tradicionais.

2.2 Modelos de Precificação

2.2.1 Modelos de Black-Scholes, Black e Garman-Kohlhagen

O modelo de Black-Scholes (1973), baseado nas teorias de Fischer Black, Myron Scholes e Robert Merton, para precificação de opções européias de compra ou venda de ações sem dividendos, é amplamente utilizado devido a sua simplicidade e velocidade de resposta. O modelo de BS, entretanto, requer para sua aplicação adequada que diversas hipóteses sejam respeitadas.

Uma das principais hipóteses do modelo de BS supõe que o comportamento dos retornos percentuais de uma determinada ação em um pequeno espaço de tempo segue uma distribuição normal, ou seja, que o preço da ação segue uma distribuição lognormal.

Portanto, se assumirmos que a volatilidade σ da ação é constante e definirmos μ e S como o retorno esperado e o preço atual de uma determinada

ação sem dividendos e δt como um espaço infinitesimal de tempo, estaremos supondo que

$$\delta S = \mu \cdot S \cdot \delta t + \sigma \cdot S \cdot \delta z \quad (2.2.1.1)$$

Onde z é uma variável que segue um processo de Wiener.

Desenvolvendo δS , obtemos

$$\begin{aligned} \delta S &= \mu \cdot S \cdot \delta t + \sigma \cdot S \cdot (\varepsilon \cdot \sqrt{\delta t}) \\ \frac{\delta S}{S} &= \mu \cdot \delta t + \sigma \cdot \varepsilon \cdot \sqrt{\delta t} \\ \frac{\delta S}{S} &\sim N(\mu \cdot \delta t, \sigma \cdot \sqrt{\delta t}) \end{aligned} \quad (2.2.1.2)$$

Onde $N(a,b)$ representa uma distribuição normal de média a e desvio padrão b e ε corresponde a uma variável aleatória que segue uma distribuição normal padronizada.

Pelo Lema de Itô, sabemos que uma variável f que seja função apenas de S e t , onde δS segue o processo descrito anteriormente, será equivalente a

$$\delta f = \left(\frac{\partial f}{\partial S} \cdot \mu \cdot S + \frac{\partial f}{\partial t} + \frac{1}{2} \cdot \frac{\partial^2 f}{\partial S^2} \cdot \sigma^2 \cdot S^2 \right) \cdot \delta t + \left(\frac{\partial f}{\partial S} \cdot \sigma \cdot S \right) \cdot \delta z \quad (2.2.1.3)$$

Observe que os processos de Wiener encontrados em δS e δf são idênticos.

Portanto, considerando que S representa o preço de uma ação sem dividendos e que f representa o preço de um derivativo (isto é, de um ativo ou instrumento financeiro cujo preço deriva ou depende) de S e t , podemos compor um portfólio P tal que $P = \frac{\partial f}{\partial S} \cdot S - f$ (ou seja, composto por $\partial f / \partial S$ unidades compradas da ação e por uma unidade vendida do derivativo).

Considerando as equações definidas para δS e δf , observamos que uma variação infinitesimal no valor do portfólio pode ser expressa por

$$\begin{aligned} \delta P &= \frac{\partial f}{\partial S} \cdot \delta S - \delta f \\ \delta P &= \frac{\partial f}{\partial S} \cdot (\mu \cdot S \cdot \delta t + \sigma \cdot S \cdot \delta z) - \left[\left(\frac{\partial f}{\partial S} \cdot \mu \cdot S + \frac{\partial f}{\partial t} + \frac{1}{2} \cdot \frac{\partial^2 f}{\partial^2 S} \cdot \sigma^2 \cdot S^2 \right) \cdot \delta t + \left(\frac{\partial f}{\partial S} \cdot \sigma \cdot S \right) \cdot \delta z \right] \\ \delta P &= - \left(\frac{\partial f}{\partial t} + \frac{1}{2} \cdot \frac{\partial^2 f}{\partial^2 S} \cdot \sigma^2 \cdot S^2 \right) \cdot \delta t \end{aligned} \quad (2.2.1.4)$$

Como δP não apresenta nenhuma variável aleatória, se assumirmos uma taxa de juros r constante, podemos considerar que $\delta P = r \cdot P \cdot \delta t$ onde concluímos que

$$\begin{aligned} - \left(\frac{\partial f}{\partial t} + \frac{1}{2} \cdot \frac{\partial^2 f}{\partial^2 S} \cdot \sigma^2 \cdot S^2 \right) \cdot \delta t &= r \cdot P \cdot \delta t \\ - \left(\frac{\partial f}{\partial t} + \frac{1}{2} \cdot \frac{\partial^2 f}{\partial^2 S} \cdot \sigma^2 \cdot S^2 \right) \cdot \delta t &= r \cdot \left(\frac{\partial f}{\partial S} \cdot S - f \right) \cdot \delta t \\ r \cdot f &= \frac{\partial f}{\partial t} + \frac{\partial f}{\partial S} \cdot r \cdot S + \frac{1}{2} \cdot \frac{\partial^2 f}{\partial^2 S} \cdot \sigma^2 \cdot S^2 \end{aligned} \quad (2.2.1.5)$$

Essa é a equação diferencial de BS que, dependendo das condições de contorno (isto é, das características) do derivativo envolvido e da observância das hipóteses do modelo de BS, possibilita a precificação do derivativo.

Por exemplo, no caso de uma opção de compra do tipo europeu (isto é, cujo exercício antecipado não é permitido) de uma ação que não paga dividendos, podemos deduzir, segundo Hull (2002), o preço da opção de compra através de uma precificação neutra ao risco obtendo:

$$c = S \cdot N(d_1) - K \cdot e^{-rT} \cdot N(d_2) \quad (2.2.1.6)$$

$$d_1 = \frac{\ln(S/K) + (r + \sigma^2/2) \cdot T}{\sigma \cdot \sqrt{T}} \quad (2.2.1.7)$$

$$d_2 = d_1 - \sigma \cdot \sqrt{T} \quad (2.2.1.8)$$

Onde S representa o preço à vista da ação, K o preço de exercício da opção, T o prazo até o vencimento, σ a volatilidade constante da ação e r a taxa de juros continuamente composta e constante.

Analogamente, através da precificação neutra ao risco, sabemos o preço de uma opção de venda desta ação corresponde a

$$p = K \cdot e^{-rT} \cdot N(-d_2) - S \cdot N(-d_1) \quad (2.2.1.9)$$

Em 1976, Fischer Black estendeu a aplicação das equações acima à precificação de opções de futuros obtendo:

$$c = e^{-rT} \cdot [F_0 \cdot N(d_1) - K \cdot N(d_2)] \quad (2.2.1.10)$$

$$p = e^{-rT} \cdot [K \cdot N(-d_2) - F_0 \cdot N(-d_1)] \quad (2.2.1.11)$$

$$d_1 = \frac{\ln\left(\frac{F_0}{K}\right) + \frac{\sigma^2}{2} \cdot T}{\sigma \cdot \sqrt{T}} \quad (2.2.1.12)$$

$$d_2 = d_1 - \sigma \cdot \sqrt{T} \quad (2.2.1.13)$$

Onde F_0 corresponde ao valor presente do respectivo contrato de Futuro do Dólar.

Em 1983, Mark Garman e Steven Kohlhagen modificaram essas equações para possibilitar a precificação de opções de moedas. No modelo de Garman-Kohlhagen (GK), os preços das opções de compra e venda são dados por

$$c = S \cdot e^{-r_{int}T} N(d_1) - K \cdot e^{-r_{dom}T} \cdot N(d_2) \quad (2.2.1.14)$$

$$p = K \cdot e^{-r_{dom}T} \cdot N(-d_2) - S \cdot e^{-r_{int}T} \cdot N(-d_1) \quad (2.2.1.15)$$

$$d_1 = \frac{\ln\left(\frac{S}{K}\right) + \left(r_{dom} - r_{int} + \frac{\sigma^2}{2}\right) \cdot T}{\sigma \cdot \sqrt{T}} \quad (2.2.1.16)$$

$$d_2 = d_1 - \sigma \cdot \sqrt{T} \quad (2.2.1.17)$$

Onde r_{int} corresponde à taxa de juros livre de risco internacional, continuamente composta, e r_{dom} , à taxa de juros livre de risco doméstica, também continuamente composta.

Devemos lembrar que, adicionalmente ao comportamento lognormal do preço do ativo-base e as taxas de juros livres de risco e constantes, os modelos de BS e GK assumem também as seguintes hipóteses:

- Inexistência de exercício antecipado;
- Ausência de custos de transação;
- Ausência de oportunidades de arbitragem;
- Continuidade dos preços;
- Divisibilidade total dos ativos;

2.2.2 Árvores Binomiais e Trimoniais

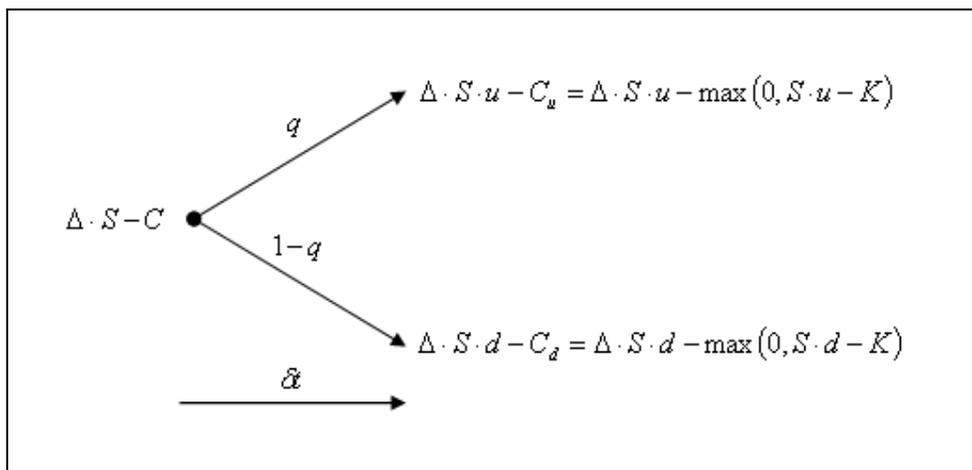
O modelo Binomial de precificação de opções, introduzido por Sharpe (1978) e posteriormente desenvolvido por Cox, Ross and Rubinstein (1979) e por Rendleman e Bartter (1979) é, segundo Hull (2002), um modelo flexível e conhecido, que possibilita a precificação de opções com características mais sofisticadas como, por exemplo, exercícios antecipados e barreiras.

A principal hipótese do modelo Binomial assume que o comportamento dos preços de um determinado ativo-base segue uma distribuição binomial, ou seja, que a cada período δt , o preço S do ativo-base pode somente ou aumentar para $S \cdot u$ ou diminuir para $S \cdot d$, tal que $0 < d < 1 + r < u$, onde r é representa a taxa livre de risco.

Assumindo que esta hipótese seja satisfeita, adotemos como ativo-base uma ação que, a priori, não paga dividendos e construamos um portfólio composto por uma posição comprada em Δ ações, de preço S , e uma posição vendida em uma opção de compra, de preço C , desta ação.

Sabemos que o preço da ação, após o período δt , corresponderá ou a $S \cdot u$, com probabilidade q ($0 < q < 1$), ou a $S \cdot d$, com probabilidade $1 - q$, e que o preço da opção será equivalente, respectivamente, a $\max(0; S \cdot u - K)$ ou a $\max(0; S \cdot d - K)$, onde K representa o preço de exercício da opção. A figura 2.2.2.1 representa o comportamento deste portfólio para o período δt .

Figura 2.2.2.1: Comportamento Binomial do Portfólio $\Delta \cdot S - C$ para um Único Período



Se adotarmos

$$\Delta = \frac{C_u - C_d}{(u - d) \cdot S} \quad (2.2.2.1)$$

De tal modo que

$$\Delta \cdot S \cdot u - C_u = \Delta \cdot S \cdot d - C_d \quad (2.2.2.2)$$

Obteremos um portfólio livre de risco.

Sabemos que um portfólio livre de risco deve, na ausência de oportunidades de arbitragem, ser remunerado a taxa livre de risco. Portanto, notamos que o valor futuro deste portfólio, após o período δt , deverá ser equivalente a

$$(\Delta \cdot S - C) \cdot e^{r \cdot \delta t} \quad (2.2.2.3)$$

Conseqüentemente, observamos que

$$\Delta \cdot S \cdot u - C_u = (\Delta \cdot S - C) \cdot e^{r \cdot \delta t} \quad (2.2.2.4)$$

$$\left(\frac{C_u - C_d}{(u-d) \cdot S} \right) \cdot S \cdot u - C_u = \left[\left(\frac{C_u - C_d}{(u-d) \cdot S} \right) \cdot S - C \right] \cdot e^{r \cdot \delta t} \quad (2.2.2.5)$$

$$\left(\frac{C_u - C_d}{(u-d)} \right) \cdot u - C_u = \left[\left(\frac{C_u - C_d}{(u-d)} \right) - C \right] \cdot e^{r \cdot \delta t} \quad (2.2.2.6)$$

$$\left(\frac{C_u - C_d}{(u-d)} \right) \cdot u - C_u = \left[\left(\frac{C_u - C_d}{(u-d)} \right) - C \right] \cdot e^{r \cdot \delta t} \quad (2.2.2.7)$$

$$C \cdot e^{r \cdot \delta t} = C_u \cdot \left(-\frac{u}{u-d} + 1 + \frac{e^{r \cdot \delta t}}{u-d} \right) + C_d \cdot \left(\frac{u}{u-d} - \frac{e^{r \cdot \delta t}}{u-d} \right) \quad (2.2.2.8)$$

$$C \cdot e^{r \cdot \delta t} = C_u \cdot \left(\frac{-u+u-d}{u-d} + \frac{e^{r \cdot \delta t}}{u-d} \right) + C_d \cdot \left(\frac{u}{u-d} - \frac{e^{r \cdot \delta t}}{u-d} \right) \quad (2.2.2.9)$$

$$C = \left(C_u \cdot \frac{e^{r \cdot \delta t} - d}{u-d} + C_d \cdot \frac{u - e^{r \cdot \delta t}}{u-d} \right) \cdot e^{-r \cdot \delta t} \quad (2.2.2.10)$$

Onde, adotando

$$p = \frac{e^{r \cdot \delta t} - d}{u-d} \quad (2.2.2.11)$$

Obtemos

$$C = [p \cdot C_u + (1-p) \cdot C_d] \cdot e^{-r \cdot \delta t} \quad (2.2.2.12)$$

Analisando a dedução da equação 2.2.2.12, percebemos que C independe das probabilidades subjetivas q e $1-q$ do investidor associadas, respectivamente, ao aumento e à redução do preço da ação.

Essa característica é fundamental pois indica que, respeitadas as hipóteses do modelo, podemos assumir na precificação das opções que os investidores são indiferentes ao risco e que, portanto, a remuneração esperada para quaisquer ativos corresponderá a taxa livre de risco.

Quando estendemos a análise binomial para N períodos, o comportamento da árvore depende dos valores escolhidos para u , d e p que são selecionados de modo a aproximar a média e a variância do passeio aleatório discreto da árvore binomial àquelas do passeio aleatório contínuo do ativo-base.

Em relação à média, lembrando que o retorno esperado de um portfólio livre de risco deve ser equivalente a taxa livre de risco, observamos que

$$E[P] = E[(\Delta \cdot S - C) \cdot e^{r \cdot \delta}] = E[(\Delta \cdot S - C) \cdot p \cdot u + (\Delta \cdot S - C) \cdot (1-p) \cdot d]$$

Tal que

$$e^{r \cdot \delta} = p \cdot u + (1-p) \cdot d \quad (2.2.2.13)$$

Em relação à variância, notamos que

$$E[P^2] - E[P]^2 = p \cdot u^2 + (1-p) \cdot d^2 + [p \cdot u + (1-p) \cdot d]^2 = \sigma^2 \cdot \delta \quad (2.2.2.14)$$

Onde substituindo p segundo a equação 2.2.2.11

$$\begin{aligned} \frac{e^{r \cdot \delta} - d}{u - d} \cdot u^2 + \left(1 - \frac{e^{r \cdot \delta} - d}{u - d}\right) \cdot d^2 + \left[\frac{e^{r \cdot \delta} - d}{u - d} \cdot u + \left(1 - \frac{e^{r \cdot \delta} - d}{u - d}\right) \cdot d\right]^2 &= \sigma^2 \cdot \delta \\ \frac{e^{r \cdot \delta} - d}{u - d} \cdot u^2 + \frac{u - e^{r \cdot \delta}}{u - d} \cdot d^2 + \left[\frac{e^{r \cdot \delta} - d}{u - d} \cdot u + \frac{u - e^{r \cdot \delta}}{u - d} \cdot d\right]^2 &= \sigma^2 \cdot \delta \\ \frac{e^{r \cdot \delta} (u^2 - d^2) - d \cdot u (u - d)}{u - d} + \left[\frac{e^{r \cdot \delta} (u - d) - d \cdot u + u \cdot d}{u - d}\right]^2 &= \sigma^2 \cdot \delta \end{aligned}$$

Encontramos

$$e^{r\cdot\delta}(u+d) - d\cdot u + e^{2r\cdot\delta} = \sigma^2 \cdot \delta \quad (2.2.2.15)$$

As equações 2.2.2.13 e 2.2.2.15, em conjunto com uma terceira equação arbitrária, constituem um sistema de equações que possibilita identificar os parâmetros u , d e p .

Cox, Ross e Rubinstein (1979), Jarrow e Rudd (1983), Hull e White (1988), Trigeorgis (1991) e Tian (1993) são exemplos de abordagens distintas para o cálculo destes parâmetros e para a construção da árvore binomial. Pelo modelo de CRR, por exemplo, impomos $d = 1/u$ de modo que

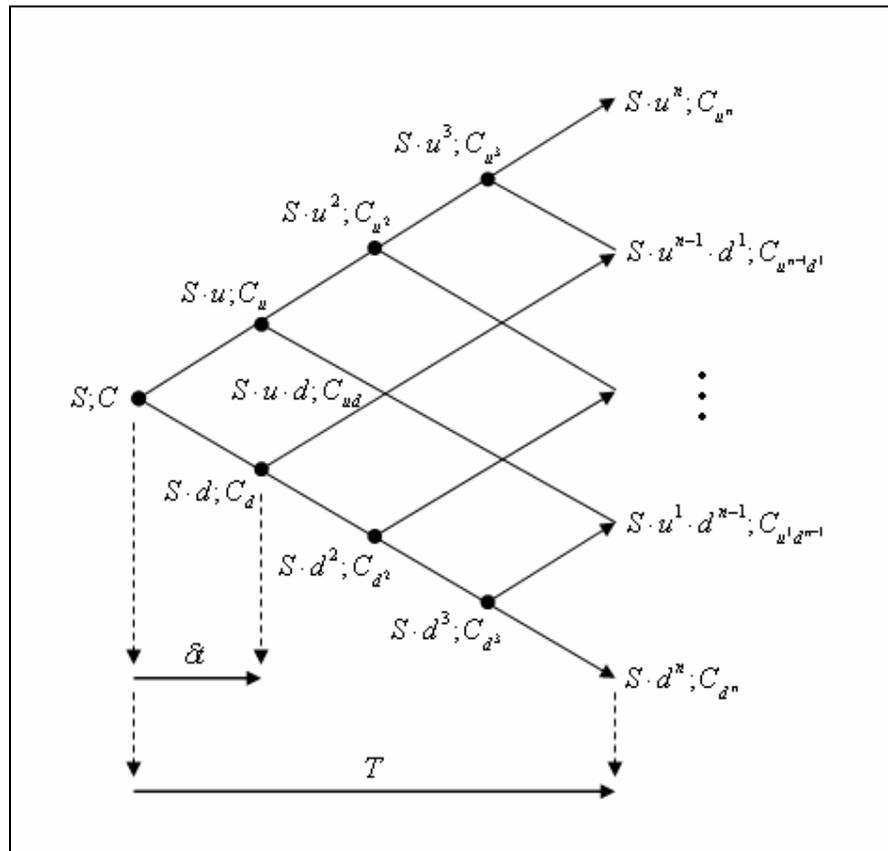
$$u = e^{\sigma\sqrt{\delta}} \quad (2.2.2.16)$$

$$d = e^{-\sigma\sqrt{\delta}} \quad (2.2.2.17)$$

$$p = \frac{e^{r\cdot\delta} - d}{u - d} \quad (2.2.2.18)$$

A escolha dos parâmetros u , d e p segundo o modelo de CRR é particularmente interessante porque, segundo Nawalkha e Chambers (1995), assegura a independência do modelo binomial de precificação de opções das preferências a risco dos investidores, mesmo em um passeio aleatório discreto.

A figura 2.2.2.2 apresenta o comportamento do modelo Binomial para N períodos.

Figura 2.2.2.2: Comportamento Binomial do Portifólio $\Delta \cdot S - C$ para N Períodos

Analisando a figura 2.2.2.2, observamos que se o preço de exercício da opção for conhecido em T , podemos encontrar seu preço em um nó qualquer da árvore recursivamente, partindo do último período.

No caso de uma opção americana de compra, por exemplo, Clewlow e Strickland (1998) demonstraram que

$$C_{i,j} = \max\left(e^{-r \cdot \Delta t} \left[p \cdot C_{i+1,j+1} + (1-p) \cdot C_{i+1,j} \right]; K - S_{i,j} \right) \quad (2.2.2.19)$$

Onde i representa o número de períodos Δt observados e j o número de elevações menos reduções ocorridas no preço do ativo-base, ambos a partir do instante zero.

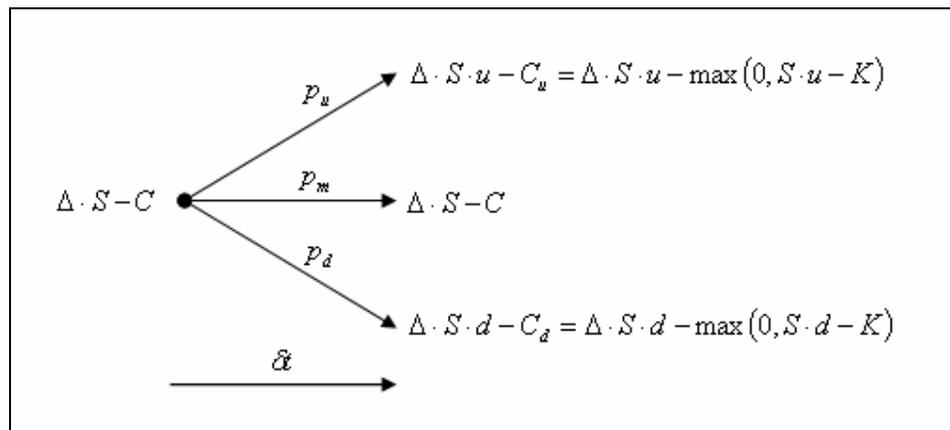
As abordagens de precificação neutra ao risco e cálculo recursivo dos preços das opções que analisamos previamente também podem ser utilizadas na construção dos modelos Trinomiais de precificação de opções.

Inicialmente proposto por Parkinson (1977) e posteriormente desenvolvido por Boyle (1986), o modelo Trinomial é uma generalização do modelo Binomial e um caso particular do modelo de Diferenças Finitas Explícitas.

A velocidade de convergência, a possibilidade de reversão à média do ativo-base e a facilidade de manipulação das volatilidades locais em cada nó são algumas vantagens do modelo Trinomial em relação ao modelo Binomial.

A figura 2.2.2.3 apresenta o comportamento do modelo Trinomial para um único período Δt quando adotamos o aumento, a redução e a manutenção do preço do ativo-base como únicas possibilidades de variação do mesmo.

Figura 2.2.2.3: Comportamento Trinomial do Portifólio $\Delta \cdot S - C$ para Um Único Período



Analogamente à dedução realizada para o modelo Binomial, a relação entre os parâmetros do modelo Trinomial deve assegurar que a média e a variância do passeio aleatório discreto da árvore trinomial sejam equivalentes àquelas do passeio aleatório contínuo do ativo-base.

Portanto, os parâmetros u , d e p_u , p_m e p_d devem satisfazer as equações

$$E[P] = p_u \cdot u + p_m \cdot 0 + p_d \cdot d = e^{r \cdot \delta t} \quad (2.2.2.20)$$

$$E[P^2] - E[P]^2 = p_u \cdot u^2 + p_m \cdot 0 + p_d \cdot d^2 + [p_u \cdot u + p_d \cdot d]^2 = \sigma^2 \cdot \delta t \quad (2.2.2.21)$$

$$p_u + p_m + p_d = 1 \quad (2.2.2.22)$$

De modo que, se adotarmos

$$u = e^{\sigma \sqrt{3 \delta t}} \quad (2.2.2.23)$$

$$d = \frac{1}{u} \quad (2.2.2.24)$$

E desconsiderarmos os termos δt de segunda ordem ou superiores, obteremos, segundo Hull (2002)

$$p_u = \sqrt{\frac{\delta t}{12 \sigma^2}} \left(r - \frac{\sigma^2}{2} \right) + \frac{1}{6} \quad (2.2.2.25)$$

$$p_m = \frac{2}{3} \quad (2.2.2.26)$$

$$p_d = \sqrt{\frac{\delta t}{12 \sigma^2}} \left(r - \frac{\sigma^2}{2} \right) - \frac{1}{6} \quad (2.2.2.27)$$

A escolha $u \geq e^{\sigma \sqrt{3 \delta t}}$, segundo Clewlow e Strickland (1998), garante que as probabilidades p_u , p_m e p_d sejam positivas e que as condições de estabilidade e convergência sejam atendidas.

Clewlow e Strickland (1998) também demonstram que, se utilizarmos $\ln S$ no lugar de S , podemos transformar as equações 2.2.2.20 e 2.2.2.21 de modo que

$$\delta x = \sigma \sqrt{3 \delta t} \quad (2.2.2.28)$$

$$p_u \cdot \Delta x + p_m \cdot 0 + p_d \cdot (-\Delta x) = \left(r - \frac{\sigma^2}{2} \right) \delta t \quad (2.2.2.29)$$

$$p_u \cdot \Delta x^2 + p_m \cdot 0 + p_d \cdot (-\Delta x)^2 = \sigma^2 \cdot \delta t + \left(r - \frac{\sigma^2}{2} \right)^2 \delta t^2 \quad (2.2.2.30)$$

Segundo este modelo, o preço de uma opção americana de compra em um nó qualquer da árvore pode ser encontrado recursivamente por

$$C_{i,j} = \max\left(e^{-r\Delta t} \left[p_u \cdot C_{i+1,j+1} + p_m \cdot C_{i+1,j} + p_d \cdot C_{i+1,j-1} \right]; K - S_{i,j} \right) \quad (2.2.2.31)$$

Variações dos modelos binomiais e trinomiais descritos previamente podem ser encontradas em Boyle, Evnine e Gibbs (1989), Madan, Milne e Shefrin (1989), Rubinstein (1994) e Figlewski e Gao (1999).

2.2.3 Simulações de Monte Carlo

O modelo de simulações de Monte Carlo é um modelo flexível e robusto para a precificação de opções que simula M vezes o comportamento de uma ou mais variáveis aleatórias (ou quase aleatórias) estimando, através da média dos valores encontrados nas simulações, o preço do derivativo em análise.

Hull (2002) comenta que o modelo de MC apresenta algumas vantagens sobre outros modelos na precificação de opções que envolvam três ou mais variáveis estocásticas ou cujo comportamento dos processos estocásticos sejam mais complexos.

Entretanto, ele destaca como limitações relevantes do modelo o número elevado de simulações necessário para reduzir a estimativa do erro e a dificuldade no tratamento de eventuais exercícios antecipados.

Para encontrar a relação entre o número de simulações e a precisão da estimativa, suponhamos uma variável aleatória x com média μ_x , cujo valor simulamos M vezes.

Sabemos que

$$\hat{x} = \frac{1}{M} \sum_{j=1}^M x_j \quad (2.2.3.1)$$

E que

$$\hat{s}^2 = \frac{1}{M-1} \sum_{j=1}^M (x_j - \hat{x})^2 \quad (2.2.3.2)$$

Lembrando que, pelo Teorema do Limite Central, para M grande,

$$\frac{\hat{x} - \mu_x}{\frac{\hat{s}}{\sqrt{M}}} \sim N(0;1) \quad (2.2.3.3)$$

Concluimos que, para um dado nível de confiança,

$$\mu_x = \hat{x} \pm \alpha_{NC} \cdot \frac{\hat{s}}{\sqrt{M}} \quad (2.2.3.4)$$

Portanto, para reduzir o erro da estimativa \hat{x} precisamos aumentar o número de simulações M ou reduzir a variância \hat{s}^2 da variável x . Como o tempo de execução está diretamente relacionado ao número de simulações, foram desenvolvidas técnicas de redução da variância para aperfeiçoar o modelo de MC.

Por exemplo, a técnica da variável antitética, proposta por Boyle (1977), envolve a substituição da simulação de uma determinada variável aleatória f , de esperança $E[f] = \mu$, pela simulação da média aritmética de duas variáveis aleatórias f_1 e f_2 , também de esperança $E[f_1] = E[f_2] = \mu$, que apresentem elevada correlação negativa.

Analisando a esperança e a variância de $\frac{f_1 + f_2}{2}$

$$E\left[\frac{f_1 + f_2}{2}\right] = \frac{E[f_1] + E[f_2]}{2} = E[f] \quad (2.2.3.5)$$

$$\text{Var}\left(\frac{f_1 + f_2}{2}\right) = \frac{\text{Var}(f_1)}{4} + \frac{\text{Var}(f_2)}{4} + \frac{\text{Cov}(f_1; f_2)}{2} \quad (2.2.3.6)$$

Observamos que se f_1 e f_2 forem escolhidos adequadamente, podemos obter $Var\left(\frac{f_1 + f_2}{2}\right) \ll Var(f)$, reduzindo significativamente o número de simulações necessárias para alcançar um determinado nível de precisão.

Outra técnica desenvolvida para redução de variância, também proposta por Boyle (1977), envolve a utilização de uma variável de controle. Esta técnica pode ser empregada quando dispomos de um derivativo, similar àquele que desejamos analisar, que ofereça uma solução analítica.

Por exemplo, sejam f_1^{MC} a solução estimada pelo modelo de MC para o derivativo sem solução analítica e f_2 e f_2^{MC} , respectivamente, a solução analítica e estimada pelo modelo de MC do derivativo similar. Podemos calcular f_1 através da equação

$$f_1 = f_1^{MC} + (f_2 - f_2^{MC}) \quad (2.2.3.7)$$

Onde $f_2 - f_2^{MC}$ se comporta como uma variável de controle.

Como

$$Var(f_2) = Cov(f_1^{MC}; f_2) = Cov(f_2^{MC}; f_2) = 0 \quad (2.2.3.8)$$

Observamos que

$$Var(f_1) = Var(f_1^{MC}) + Var(f_2^{MC}) - 2Cov(f_1^{MC}; f_2^{MC}) \quad (2.2.3.9)$$

De modo que, se escolhermos f_2 apropriadamente, podemos melhorar a estimativa de f_1 reduzindo sua variância.

Adicionalmente as técnicas de redução de variância mencionadas, Boyle, Broadie e Glasserman (1997), Broadie, Glasserman e Jain (1997) e Hull (2002) apresentam outros métodos para melhorar o desempenho do modelo de MC como,

por exemplo, a amostragem por importância, a amostragem estratificada, o emparelhamento de momentos e as seqüências quase-aleatórias.

2.2.4 Redes Neurais Artificiais otimizadas por Algoritmos Genéticos

2.2.4.1 Redes Neurais Artificiais

Segundo Haykin (1999), uma rede neural artificial (RNA) é um modelo de inferência que busca representar o processo de raciocínio do cérebro humano durante a execução de alguma atividade.

As RNAs são constituídas por unidades mais elementares de processamento denominadas neurônios, responsáveis pelo cálculo de determinadas funções matemáticas que, usualmente, não são lineares.

Os neurônios são agrupados em uma ou mais camadas e interligados através de um grande número de conexões, as quais são associadas a pesos que, como veremos posteriormente, representam o conhecimento de uma RNA.

Essa estrutura permite a uma RNA realizar o processamento não linear e paralelo de um conjunto de informações oferecendo, segundo Braga *et al.* (2000), um desempenho superior ao dos modelos paralelos convencionais tais como Clustering e Grade de Processamento.

As figuras 2.2.4.1.1 e 2.2.4.1.2 apresentam, respectivamente, um modelo não linear de neurônio (Haykin, 1999) e algumas arquiteturas comuns de RNAs (Braga *et al.* 2000).

Figura 2.2.4.1.1: Modelo de um Neurônio Não Linear

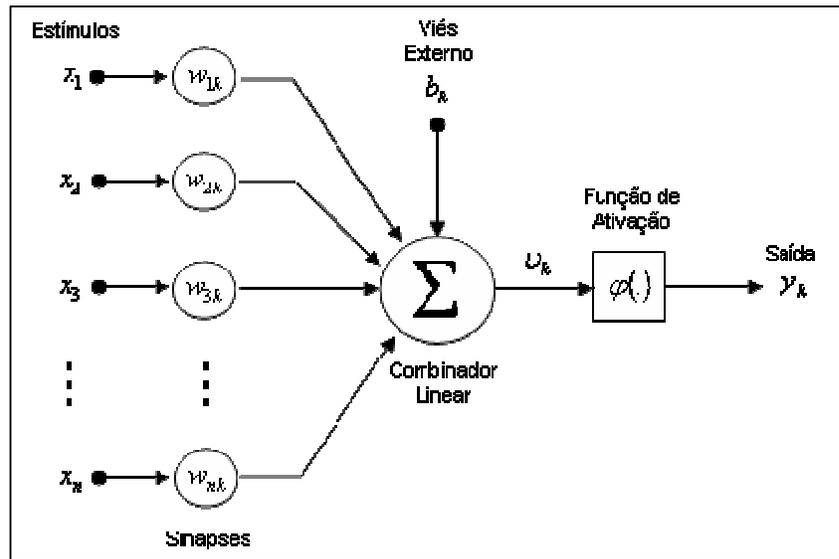
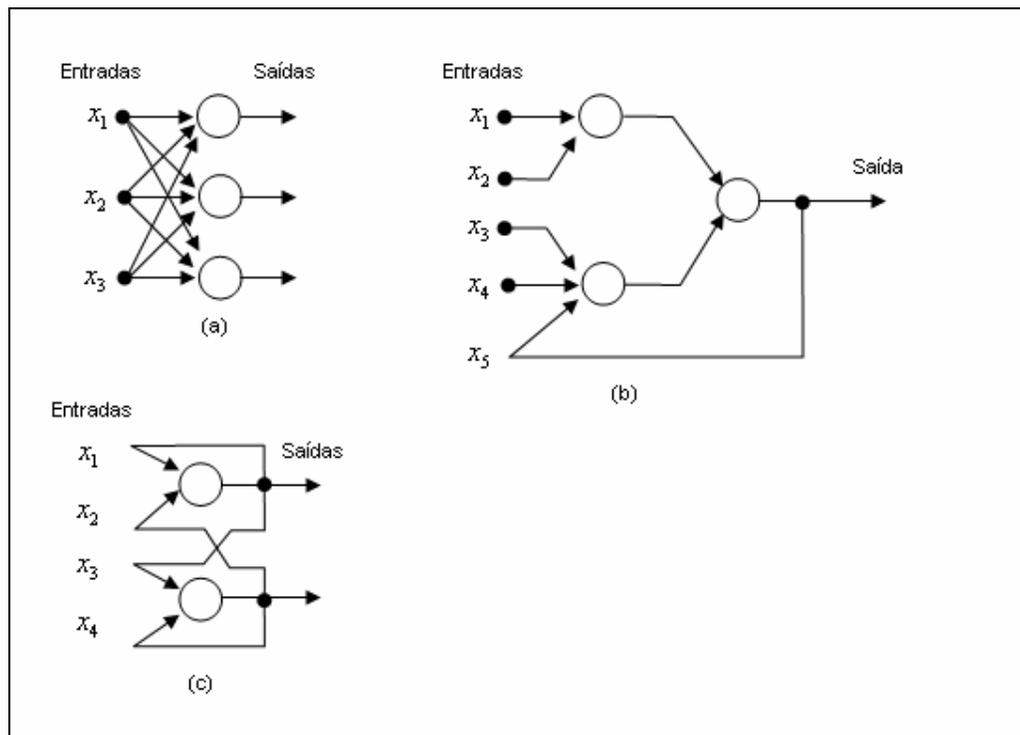


Figura 2.2.4.1.2: Exemplos de Arquiteturas de RNAs



No modelo de neurônio apresentado na figura 2.2.4.1.1, observamos três componentes fundamentais:

- As sinapses ou conexões de entrada x_n e seus pesos individuais w_{kn} que determinam a influência de um determinado estímulo externo x no neurônio k ;
- Um combinador linear das informações transmitidas pelas sinapses e de um viés externo (denominado limiar de ativação) e ;
- Uma função de ativação $\varphi(\cdot)$ responsável pela normalização da amplitude da informação de saída y_k onde

$$y_k = \varphi(v_k) = \varphi\left(\sum_{j=1}^m w_{kj} \cdot x_j + b_k\right) \quad (2.2.4.1.1)$$

As funções de ativação, segundo Haykin (1999), podem ser divididas em três grupos básicos: degrau, rampa e sigmoideal sendo este último o mais utilizado na construção de RNAs. A equação 2.2.4.1.2 apresenta um exemplo de uma função sigmoideal logística:

$$y_k = \varphi(v) = \frac{1}{1 + e^{-a \cdot v}} \quad (2.2.4.1.2)$$

Como exemplificado na figura 2.2.4.1.2, os neurônios podem ser combinados de diversas formas, obtendo diferentes arquiteturas de RNAs. Os parâmetros da arquitetura de uma RNA compreendem a quantidade de camadas da rede, a quantidade de neurônios em cada camada, o tipo e o peso das conexões entre os neurônios e a topologia da RNA.

Segundo Haykin (1999), as arquiteturas das RNA podem ser classificadas em três classes principais: redes acíclicas (ou feedforward) de uma única camada, redes acíclicas de múltiplas camadas e redes recorrentes.

As redes acíclicas de múltiplas camadas são particularmente interessantes porque, como demonstrado por Cybenko (1988), uma RNA com no mínimo duas camadas intermediárias é capaz de aproximar qualquer função matemática, contínua ou não.

Um caso particular das redes acíclicas de múltiplas camadas, previamente utilizado nas aplicações e estudos orientados ao mercado financeiro (Freitas e Souza, 2004), são os Multilayer Perceptrons (MLPs), cujas principais características compreendem:

- A utilização de funções de ativação não lineares e diferenciáveis, tais como as funções sigmoidais, para cada neurônio da RNA;
- A presença de pelo menos uma camada intermediária de neurônios entre as camadas de entrada e saída;
- O elevado número de sinapses (i.e., de conexões) entre os neurônios.

Haykin (1999) explica que a combinação dessas características a um processo de aprendizagem apropriado proporciona uma significativa eficiência computacional na resolução de diversos problemas.

Como veremos adiante, o processo de aprendizagem corresponde a um algoritmo de treinamento que extrai as relações relevantes dos conjuntos de informações apresentados à RNA e os armazena através de um processo iterativo de ajuste dos pesos sinápticos de suas conexões. Ao final do processo, os pesos sinápticos representam o conhecimento que a RNA adquiriu do ambiente no qual ela está atuando.

Segundo Braga *et al.* (2000), os processos de aprendizagem podem ser agrupados em processos supervisionados e não supervisionados. Os processos supervisionados diferem dos processos não supervisionados por envolverem um

agente externo responsável pela indicação das respostas ótimas associadas a um determinado conjunto de informações de entrada.

Como explicado por Haykin (1999) e Braga *et al.* (2000), dentre os diversos mecanismos de aprendizado desenvolvidos nas últimas décadas, o algoritmo de back-propagation foi particularmente importante na disseminação dos MLPs.

O back-propagation é um algoritmo de aprendizado supervisionado por correção de erro que busca identificar o mínimo global de uma superfície de erro através da análise do gradiente de seus pontos.

Para calcular o gradiente dos pontos da superfície de erro, precisamos conhecer a função que a define. Analisando a camada de saída, observamos que o erro $e_j(n)$ do neurônio j , após o processamento do n -ésimo vetor (i.e., conjunto de informações) de entrada utilizado no treinamento da RNA, é dada pela função

$$e_j(n) = d_j(n) - y_j(n) \quad (2.2.4.1.3)$$

Onde $d_j(n)$ representa a saída correta ou desejada e $y_j(n)$ representa a saída gerada pela RNA nesta iteração.

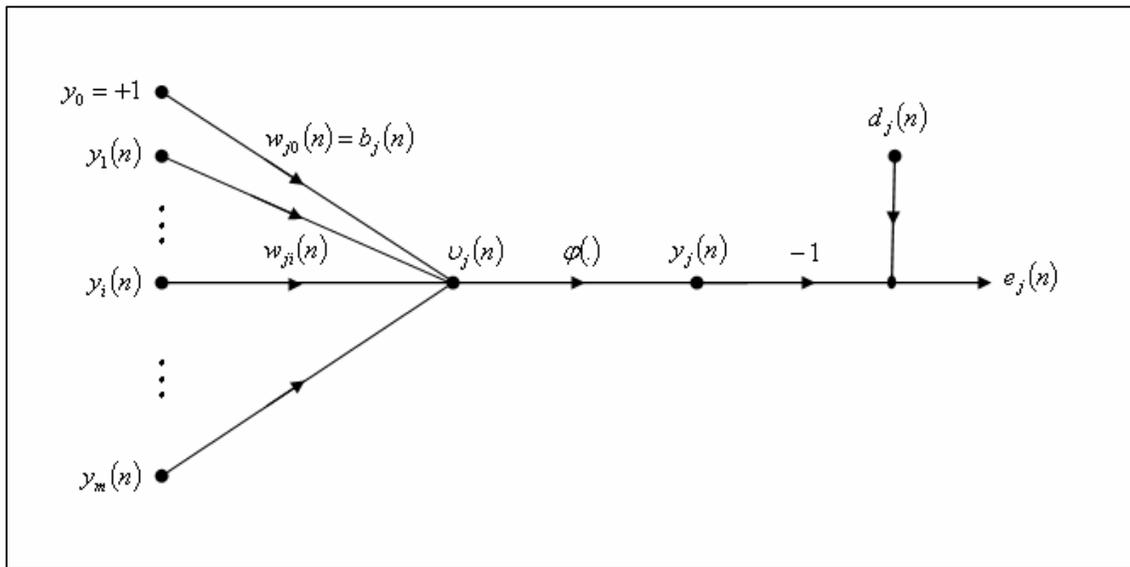
Adotando $\frac{1}{2}e_j^2(n)$ como a energia instantânea do erro do neurônio j e $\varepsilon(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n)$ como a soma da energia instantânea do erro de todos os neurônios da camada de saída da RNA, podemos definir

$$\hat{\varepsilon} = \frac{1}{N} \sum_{n=1}^N \varepsilon(n) \quad (2.2.4.1.4)$$

Onde N corresponde à quantidade de conjuntos (também denominados épocas) de vetores de entrada destinados ao treinamento da RNA e $\hat{\varepsilon}$, a média dos quadrados das energias instantâneas dos erros dos neurônios da camada de saída.

A figura 2.2.4.1.3 (Haykin, 1999), demonstra como as informações de entrada do neurônio j da camada de saída (que correspondem às informações de saída $y_i(n)$ dos neurônios da camada anterior) são combinadas e transformadas para se obter a informação de saída $y_j(n)$ e, conseqüentemente, o erro $e_j(n)$.

Figura 2.2.4.1.3: Fluxo de Informações de um Neurônio da Camada de Saída



Pela figura 2.2.4.1.3, observamos que

$$v_j(n) = \sum_{i=0}^m w_{ji}(n) \cdot y_i(n) \quad (2.2.4.1.5)$$

$$y_j(n) = \phi(v_j(n)) \quad (2.2.4.1.6)$$

$$y_j(n) = \phi\left(\sum_{i=0}^m w_{ji}(n) \cdot y_i(n)\right) \quad (2.2.4.1.7)$$

Onde m corresponde à quantidade de entradas e $\phi(\cdot)$, a função de ativação do neurônio j .

Conhecidos $y_i(n)$, notamos que $e_j(n)$ e, conseqüentemente, $\varepsilon(n)$, são funções apenas dos pesos sinápticos do neurônio j . Portanto, a variação de $\varepsilon(n)$ é proporcional às suas derivadas parciais em função dos pesos sinápticos $\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)}$ que correspondem aos gradientes de $\varepsilon(n)$.

Pela regra da cadeia, sabemos que

$$\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)} = \frac{\partial \varepsilon(n)}{\partial e_j(n)} \cdot \frac{\partial e_j(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial v_j(n)} \cdot \frac{\partial v_j(n)}{\partial w_{ji}(n)} \quad (2.2.4.1.8)$$

Onde

$$\frac{\partial \varepsilon(n)}{\partial e_j(n)} = \frac{\partial \left(\frac{1}{2} \sum_{j \in C} e_j^2(n) \right)}{\partial e_j(n)} = e_j(n) \quad (2.2.4.1.9)$$

$$\frac{\partial e_j(n)}{\partial y_j(n)} = \frac{\partial (d_j(n) - y_j(n))}{\partial y_j(n)} = -1 \quad (2.2.4.1.10)$$

$$\frac{\partial y_j(n)}{\partial v_j(n)} = \frac{\partial (\varphi(v_j(n)))}{\partial v_j(n)} = \varphi'(v_j(n)) \quad (2.2.4.1.11)$$

$$\frac{\partial v_j(n)}{\partial w_{ji}(n)} = \frac{\partial \left(\sum_{i=0}^m w_{ji}(n) \cdot y_i(n) \right)}{\partial w_{ji}(n)} = y_i(n) \quad (2.2.4.1.12)$$

E, portanto

$$\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)} = -e_j(n) \cdot \varphi'(v_j(n)) \cdot y_i(n) \quad (2.2.4.1.13)$$

Lembrando que o gradiente de $\varepsilon(n)$, $\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)}$, indica a direção na qual $\varepsilon(n)$ cresce, devemos ajustar $w_{ji}(n)$ na direção oposta para localizarmos o ponto de mínimo global da superfície de erro.

Segundo Haykin (1999), podemos ajustar $w_{ji}(n)$ através da regra de Widrow-Hoff tal que

$$\Delta w_{ji}(n) = -\eta \cdot \frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} \quad (2.2.4.1.14)$$

$$\Delta w_{ji}(n) = -\eta \cdot [-e_j(n) \cdot \varphi'(v_j(n)) \cdot y_i(n)] \quad (2.2.4.1.15)$$

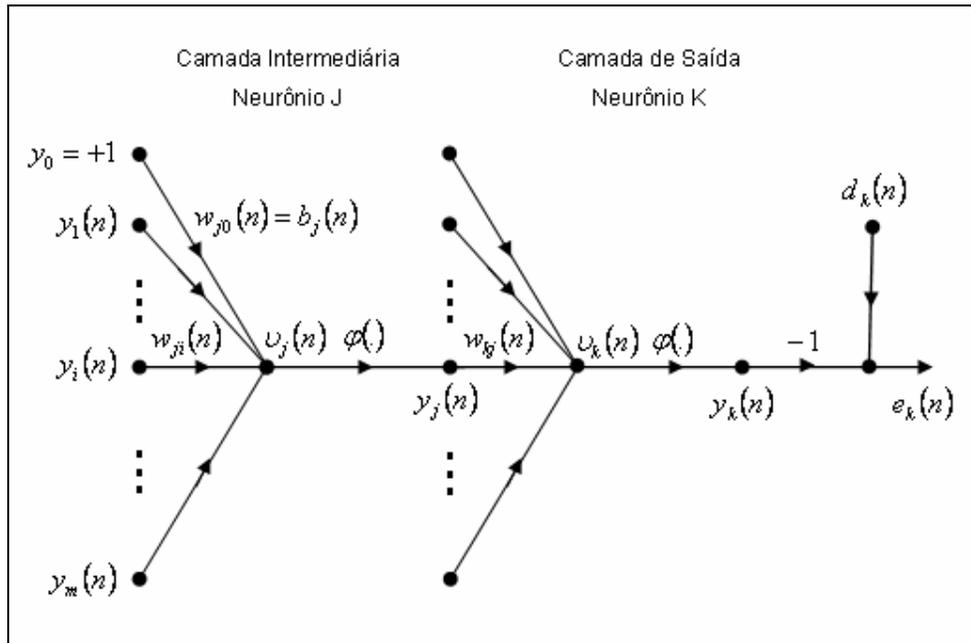
$$\Delta w_{ji}(n) = \eta \cdot \delta_j(n) \cdot y_i(n) \quad (2.2.4.1.16)$$

Onde η corresponde a taxa de aprendizado da RNA, usualmente ajustada ao longo do processo de treinamento para otimizar o processo de convergência, e $\delta_j(n)$, ao gradiente local do neurônio j .

A equação 2.2.4.1.16, conforme demonstrado, possibilita o cálculo iterativo de $\Delta w_{ji}(n)$ para os neurônios da camada de saída. Contudo, para as camadas intermediárias da RNA, o erro $e_j(n)$ não pode ser obtido diretamente, sendo necessário adaptar a equação 2.2.4.1.16.

Analisando-a, notamos que $\Delta w_{ji}(n)$ é função de η , $\delta_j(n)$ e $y_i(n)$ mas que somente $\delta_j(n)$ depende do erro $e_j(n)$. Portanto, suponhamos que o neurônio j esteja localizado em uma camada intermediária e que o neurônio k , pertença à camada de saída, conforme demonstra a figura 2.2.4.1.4 (Haykin, 1999).

Figura 2.2.4.1.4: Fluxo de Informações de um Neurônio da Camada Intermediária (Invisível)



Lembrando que

$$\delta_j(n) = e_j(n) \cdot (-1) \cdot \varphi'(v_j(n)) \quad (2.2.4.1.17)$$

$$\delta_j(n) = \frac{\partial \mathcal{E}(n)}{\partial e_j(n)} \cdot \frac{\partial e_j(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial v_j(n)} \quad (2.2.4.1.18)$$

$$\delta_j(n) = \frac{\partial \mathcal{E}(n)}{\partial v_j(n)} \quad (2.2.4.1.19)$$

E que

$$\mathcal{E}(n) = \frac{1}{2} \sum_{k \in C} e_k^2(n) \quad (2.2.4.1.20)$$

Podemos aplicar a regra da cadeia para obter

$$\delta_j(n) = - \frac{\partial \mathcal{E}(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial v_j(n)} \quad (2.2.4.1.21)$$

$$\delta_j(n) = - \frac{\partial \mathcal{E}(n)}{\partial y_j(n)} \cdot \varphi'(v_j(n)) \quad (2.2.4.1.22)$$

Pela equação 2.2.4.1.20, observamos que

$$\frac{\partial \mathcal{E}(n)}{\partial y_j(n)} = \sum_{k \in C} e_k(n) \cdot \frac{\partial e_k(n)}{\partial y_j(n)} \quad (2.2.4.1.23)$$

$$\frac{\partial \mathcal{E}(n)}{\partial y_j(n)} = \sum_{k \in C} e_k(n) \cdot \frac{\partial e_k(n)}{\partial v_k(n)} \cdot \frac{\partial v_k(n)}{\partial y_j(n)} \quad (2.2.4.1.24)$$

Adicionalmente, pela figura 2.2.4.1.4, notamos que

$$e_k(n) = d_k(n) - y_k(n) \quad (2.2.4.1.25)$$

$$e_k(n) = d_k(n) - \varphi(v_k(n)) \quad (2.2.4.1.26)$$

E que

$$v_k(n) = \sum_{j=0}^{m_k} w_{kj}(n) \cdot y_j(n) \quad (2.2.4.1.27)$$

Portanto, concluímos que

$$\frac{\partial e_k(n)}{\partial v_k(n)} = -\varphi'(v_k(n)) \quad (2.2.4.1.28)$$

$$\frac{\partial v_k(n)}{\partial y_j(n)} = w_{kj}(n) \quad (2.2.4.1.29)$$

De tal modo que

$$\frac{\partial \mathcal{E}(n)}{\partial y_j(n)} = \sum_{k \in C} e_k(n) \cdot (-\varphi'(v_k(n))) \cdot w_{kj}(n) \quad (2.2.4.1.30)$$

$$\frac{\partial \mathcal{E}(n)}{\partial y_j(n)} = -\sum_{k \in C} \delta_k(n) \cdot w_{kj}(n) \quad (2.2.4.1.31)$$

Substituindo $\frac{\partial \mathcal{E}(n)}{\partial y_j(n)}$ na equação 2.2.4.1.22, encontramos

$$\delta_j(n) = -\left[-\sum_{k \in C} \delta_k(n) \cdot w_{kj}(n) \right] \cdot \varphi'(v_j(n)) \quad (2.2.4.1.32)$$

Onde observamos que trocando $e_j(n)$ por $\sum_{k \in C} \delta_k(n) \cdot w_{kj}(n)$ podemos calcular

$\delta_j(n)$ recursivamente através dos pesos sinápticos $w_{kj}(n)$ e do gradiente local $\delta_k(n)$

das camadas posteriores (neste caso, a camada de saída).

Portanto, para camadas intermediárias, a equação 2.2.4.1.16, que calcula os ajustes $\Delta w_{ji}(n)$ necessários aos respectivos pesos sinápticos, deve utilizar a equação 2.2.4.1.32 para calcular o gradiente local $\delta_j(n)$.

Definidas as equações utilizadas no ajuste dos pesos sinápticos durante cada iteração, podemos resumir, segundo Haykin (1999), o treinamento seqüencial de uma RNA através do algoritmo de back-propagation nas seguintes etapas:

- 1.) Inicialização dos pesos sinápticos com valores aleatórios (assumindo que não existam informações prévias disponíveis) segundo uma distribuição normal com média zero e desvio padrão apropriados (i.e., que não inicializem $v_j(0)$ na região de saturação da função de ativação $\varphi(\cdot)$).
- 2.) Introdução do vetor de entrada da época em análise e cálculo do vetor de saída $y_j(n)$ da RNA. Nesta etapa, as saídas $y_j(n)$ são atualizadas sequencialmente, da camada de entrada para a camada de saída da RNA, enquanto os pesos sinápticos $w_{ji}(n)$ permanecem inalterados.
- 3.) Cálculo dos ajustes $\Delta w_{ji}(n)$ segundo as equações demonstradas previamente. Nesta etapa, os pesos sinápticos $w_{ji}(n)$ são atualizados sequencialmente, da camada de saída para a camada de entrada da RNA, enquanto as saídas $y_j(n)$ dos neurônios da RNA permanecem inalteradas.
- 4.) Repetição das etapas 2 e 3 até que os critérios de parada (discutidos posteriormente) sejam atendidos ou até que todos os vetores de entrada de todas as épocas disponíveis sejam processados.

Infelizmente, o processo delineado previamente descreve a implementação básica do algoritmo de back-propagation que proporciona uma performance fraca em termos de velocidade de resposta.

Devido a esta deficiência, certas modificações tais como a utilização de técnicas heurísticas (momento, taxa de aprendizado variável ou propagação reversa resiliente, por exemplo) ou métodos de otimização numérica (gradiente conjugado, quasi-Newton ou Levenberg-Marquardt, por exemplo) são necessárias para melhorar o desempenho da RNA.

O aprofundamento teórico destas variações do algoritmo de back-propagation, que são exaustivamente discutidas em Hagan, Demuth e Beale (1995), foge ao escopo deste trabalho. Entretanto, as seguintes características observadas por Hagan, Demuth e Beale (1995) devem ser enfatizadas:

- A convergência do algoritmo de Levenberg-Marquardt é usualmente superior para os problemas de aproximação de funções (tais como a precificação de opções) onde a RNA é composta por apenas algumas dezenas de pesos sinápticos. Infelizmente, os recursos computacionais de armazenamento exigidos para sua aplicação são consideráveis.
- O algoritmo de quasi-Newton exhibe desempenho próximo ao algoritmo de Levenberg-Marquardt, requer menos capacidade de armazenamento mas exige uma maior capacidade de processamento.
- Os algoritmos de gradiente conjugado demonstram boa performance para diferentes tipos de problemas, especialmente quando a quantidade de pesos sinápticos da RNA é elevada.
- O algoritmo de propagação reversa resiliente usualmente apresenta melhores resultados para os problemas de identificação de padrões.

Considerando as características destacadas por Hagan, Demuth e Beale (1995) e as análises de Valença (1999) e Prudêncio (2002), adotaremos o algoritmo de back-propagation de Levenberg-Marquardt em nossa análise. Posteriormente, discutiremos como combiná-lo aos algoritmos genéticos para compará-lo aos demais modelos de especificação previamente mencionados.

Adicionalmente à escolha do algoritmo apropriado de treinamento, devemos ainda considerar os seguintes aspectos finais no processo de construção da RNA:

- Análise e tratamento de um eventual overfitting (i.e., redução da capacidade de generalização devido à memorização dos ruídos) da RNA, particularmente relevante no treinamento com algoritmos híbridos como veremos posteriormente. Podemos dizer que o overfitting corresponde a uma sobre-parametrização de um modelo como, por exemplo, a utilização de um polinômio de grau demasiadamente elevado na realização de uma interpolação polinomial pode ser considerada uma situação de overfitting.

Para a redução do overfitting relacionados aos problemas de aproximação de funções, Montana e Davis (1989) e Stepniewski e Keane (1996) recomendam:

- ✓ O método de regularização Bayesiana, que consiste na alteração da função de erro a ser minimizada para que a mesma considere e penalize pesos sinápticos elevados (forçando uma suavização da resposta da RNA e reduzindo, conseqüentemente, a influência de ruídos).
- ✓ O método de parada antecipada, que compreende a divisão das informações de entrada em dois grupos distintos: um de treinamento

e outro de validação. Este último grupo é utilizado para acompanhar e encerrar o processo de treinamento quando a resposta da RNA apresentar sinais de deterioração após um período razoável de convergência. Este método é particularmente sensível a mínimos locais (comuns em superfícies complexas).

A descrição da implementação destes métodos pode ser encontrada em Montana e Davis (1989), Hagan, Demuth e Beale (1995), Stepniewski e Keane (1996) e Haykin (1999).

- Normalização das informações de entrada e saída.
- Transformação das informações de entrada utilizando a Análise de Componentes Principais (PCA) para:
 - ✓ Eliminação de outliers e;
 - ✓ Ortogonalização dos vetores de entrada.
- Inicialização adequada dos pesos sinápticos e vieses através do algoritmo de Nguyen-Widrow que reduz o tempo de treinamento e otimiza a utilização dos neurônios da RNA. A descrição deste algoritmo pode ser encontrada em Hagan, Demuth e Beale (1995).

A seguir, discutiremos as características e os comportamentos dos algoritmos genéticos (AGs) e como combiná-los às RNAs para construir um algoritmo híbrido que será posteriormente utilizado na precificação dos instrumentos derivativos analisados neste trabalho.

2.2.4.2 Algoritmos Genéticos

Os algoritmos genéticos (AGs), inicialmente propostos por Holland (1975), podem ser definidos, segundo Goldberg (1989), como algoritmos estocásticos de busca e otimização paralela baseadas no princípio Darwiniano de reprodução genética e seleção natural dos indivíduos mais aptos.

Assim como no processo evolutivo Darwiniano, o processo de otimização dos AGs para a solução ótima (ou quase-ótima) de uma função baseia-se na variação iterativa de informações mais elementares denominadas genes e cromossomos.

A cada geração (i.e., iteração), os cromossomos ou indivíduos (i.e., possíveis soluções da função) da população (i.e., conjunto de possíveis soluções simultaneamente avaliadas) sofrem alterações decorrentes dos processos de cruzamento e mutação, que discutiremos posteriormente.

Estes processos modificam os genes (i.e., variáveis que compõem uma possível solução) dos cromossomos atuais gerando novas populações que, conforme demonstrado por Goldberg (1989), Davis (1991) e Chen (2002), tendem a apresentar, estatisticamente, indivíduos mais aptos, segundo uma determinada função de aptidão.

Basicamente, o processo de otimização pelos AGs envolve:

- A definição de uma população inicial.
- A construção iterativa e adaptativa de novas gerações da população inicial que, por sua vez, compreende:
 - ✓ Um processo seletivo dos indivíduos mais qualificados da população atual, segundo uma dada função de aptidão;
 - ✓ Um processo de cruzamento onde genes dos indivíduos escolhidos pelo processo seletivo são combinados para gerar um novo indivíduo;

✓ Um processo de mutação aleatória nos padrões genéticos da população atual responsável pela diversificação dos indivíduos das novas gerações.

- A definição de critérios de parada para o processo de otimização.

A etapa de inicialização da população inicial consiste, usualmente, na escolha aleatória de uma população de cromossomos, ou seja, de possíveis soluções para a função que desejamos otimizar.

Contudo, como explicado por Chen (2002), esta escolha aleatória deve considerar a diversidade e o tamanho da população inicial que são parâmetros particularmente importantes para a velocidade e confiabilidade da convergência dos AGs. Infelizmente, os valores destes parâmetros devem ser empiricamente definidos, pois variam de acordo com o problema analisado.

O processo seletivo, que corresponde ao primeiro passo da etapa de construção de uma nova geração da população, utiliza uma função de aptidão que associa a cada indivíduo uma nota que define seu grau de participação no processo de reprodução.

Apesar de intuitiva, a participação proporcional dos indivíduos a sua respectiva nota não é recomendada por Davis (1991) e Mitchell (1996), pois permite que indivíduos com notas elevadas dominem excessivamente o processo reprodutivo reduzindo demasiadamente o potencial de diversificação da população.

Como alternativa, Davis (1996) sugere que as notas da função de adaptação sejam utilizadas apenas para ordenar os indivíduos e, posteriormente, substituídas por notas proporcionais às posições dos indivíduos conforme a equação

$$n = \frac{1}{\sqrt{P}} \quad (2.2.4.2.1)$$

Onde p corresponde à posição do indivíduo baseada na nota original e n , a nota ajustada que utilizaremos na seleção dos indivíduos mais aptos.

Calculadas as notas ajustadas, migramos (i.e., copiamos) sem alterações (mutação ou cruzamento) os indivíduos de elite (i.e., os 5% a 10% melhores dentre os mais aptos) para a nova geração da população para otimizar o processo de convergência.

Dentre os indivíduos selecionados remanescentes, utilizamos um processo estocástico, onde a probabilidade de seleção de cada indivíduo é proporcional à sua nota ajustada, para eleger aqueles que participarão, uma ou mais vezes, dos processos de cruzamento e mutação.

O processo (também denominado operador) de cruzamento, segundo Chen (2002), é um mecanismo de busca que possibilita identificar novas soluções utilizando as melhores soluções previamente encontradas.

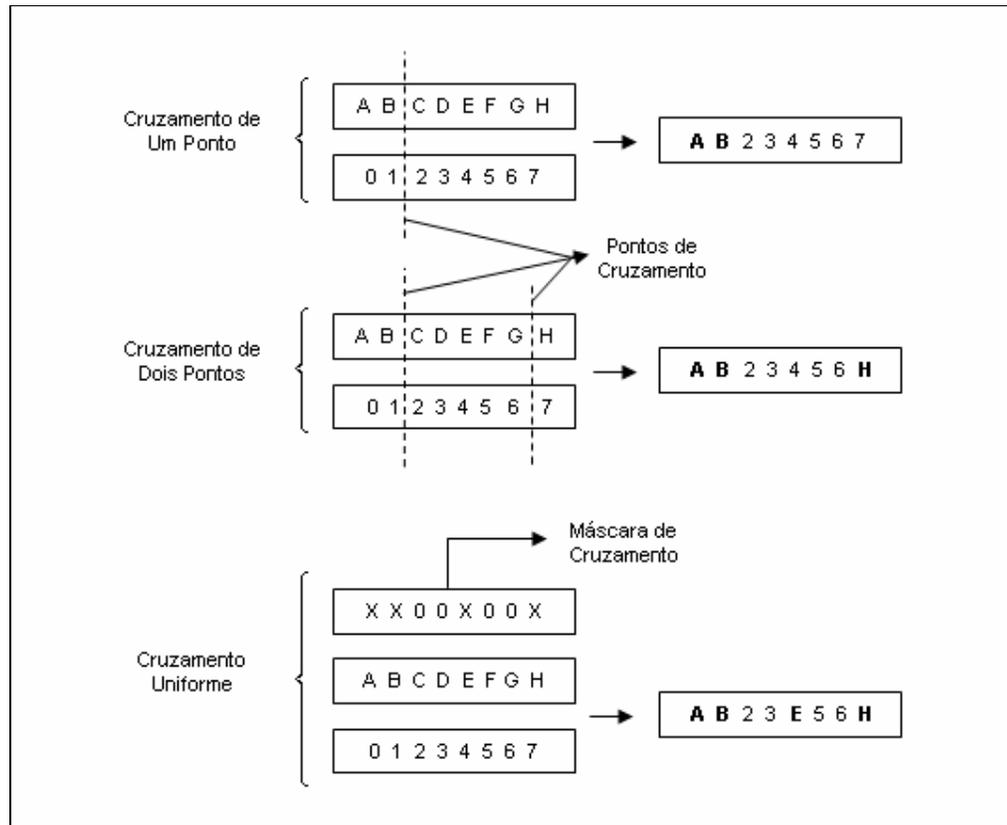
O cruzamento extrai os genes (características) dos melhores indivíduos da população atual e os recombina gerando novos indivíduos, possivelmente mais aptos.

Os genes podem ser cruzados (i.e., recombinados) de diferentes formas tais como cruzamento de ponto único, cruzamento de ponto duplo, cruzamento uniforme, entre outros.

Contudo, como explicado por Montana e Davis (1989), Koza e Rice (1991) e Zhang e Muhlenbein (1993), dependendo da codificação das informações genéticas (i.e., da forma como as informações do problema em análise serão representadas pelos genes e cromossomos), o cruzamento pode comprometer o processo de otimização, como veremos posteriormente durante a discussão dos modelos híbridos de RNAs e AGs.

A figura 2.2.4.2.1 apresenta o comportamento dos diferentes tipos de cruzamento mencionados.

Figura 2.2.4.2.1: Comportamentos dos Diferentes Tipos de Cruzamento



A cada cruzamento, novas máscaras ou pontos de cruzamento são escolhidos aleatoriamente para expandir o universo de possíveis combinações.

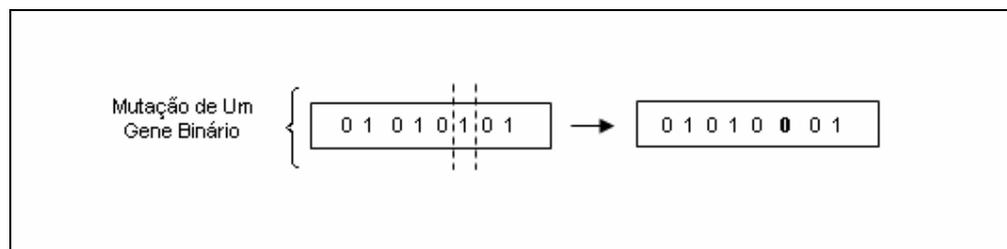
O processo (ou operador) de mutação, segundo Chen (2002), é um mecanismo que proporciona diversidade às novas gerações permitindo ao AG gerar indivíduos que não sejam apenas recombinações dos indivíduos originais.

Contudo, como o processo de mutação é computacionalmente lento, pois se assemelha a um processo de busca exaustiva pelo espaço de possíveis soluções, e

deve ser utilizado conjuntamente com outros mecanismos de busca mais eficientes tais como o cruzamento.

A figura 2.2.4.2.2 apresenta o comportamento do processo de mutação no caso de uma representação binária de genes (i.e., onde os genes podem assumir apenas os valores 0 ou 1).

Figura 2.2.4.2.2: Comportamentos do Processo de Mutação



Finalmente, a etapa de verifica o dos crit rios de parada consiste na an lise das caracter sticas da popula o atual e de sua evolu o em rela o   gera o anterior.

Usualmente, os crit rios de parada compreendem:

- O n mero de gera es (i.e., itera es) que o AG deve executar durante o processo de busca e otimiza o;
- O tempo m ximo para identifica o da solu o  tima (ou quase- tima) e;
- A precis o m nima de aptid o que a solu o encontrada pelo AG deve observar.

Quando um destes crit rios   excedido, o AG encerra sua busca.

Adicionalmente, podemos impor duas restri es suplementares:

- O n mero m ximo de gera es sem melhorias na aptid o das novas popula es;

- O tempo máximo sem melhorias na aptidão das novas populações;

Estas restrições adicionais nos ajudam a identificar uma eventual convergência prematura do AG e, conseqüentemente, uma necessidade de revisão dos parâmetros estabelecidos para o AG.

Veremos a seguir que, quando combinadas às RNAs, os AGs seguem as mesmas etapas há pouco discutidas. Entretanto, sua implementação varia segundo os parâmetros da RNA que serão otimizados e segundo a abordagem utilizada na representação destas informações para tratamento pelo AG.

2.2.4.3 Combinando Redes Neurais Artificiais e Algoritmos Genéticos

Montana e Davis (1989) e Schaffer, Whitley e Eshelman (1992) explicam que a combinação das RNAs e dos AGs pode ser utilizada para identificar determinados parâmetros das RNAs (tais como a topologia e os pesos sinápticos) e para contornar algumas limitações inerentes as RNAs (tais como a necessidade de funções deriváveis e a dificuldade no tratamento de mínimos locais).

Por exemplo, Montana e Davis (1989) propõe o uso de AGs no processo de treinamento de RNAs cujas superfícies de erro relacionadas sejam muito complexas. Nestes casos, a presença de diversos mínimos locais prejudica significativamente a utilização dos algoritmos de back-propagation.

Outro exemplo apresentado por Whitley, Starkweather e Bogart (1990), sugere a utilização de AGs para a determinação da topologia ótima da RNA, particularmente relevante no tempo de treinamento, na precisão da resposta e na determinação da capacidade de generalização da RNA.

Contudo, a utilização de algoritmos híbridos empregando AGs usualmente requer mais recursos computacionais e, conseqüentemente, maior tempo de processamento.

Por este motivo, a definição do genótipo (i.e., a representação codificada das informações) da estrutura da RNA (i.e., topologia, pesos sinápticos e liminares de ativação e demais parâmetros) que pretendemos aperfeiçoar é um aspecto fundamental na construção do modelo híbrido.

Como explicado por Dasgupta e McGregor (1992), Gruau (1993) e Gruau, Whitley e Peyatt (1996), podemos escolher entre duas abordagens para a codificação do fenótipo (i.e., a representação não codificada das informações) em seu respectivo genótipo: a direta ou a indireta.

A codificação direta, de implementação mais simples, corresponde usualmente à representação binária das informações da RNA. Contudo, como demonstrado por Maniezzo (1994), sua aplicação se restringe as RNAs com poucos parâmetros para que a eficiência do AG não seja comprometida.

A codificação indireta envolve a representação estruturada (como, por exemplo, uma árvore de neurônios) das informações da RNA possibilitando ao AG identificar, simultaneamente, a solução e a interdependência quase-ótimas contida no conjunto de informações analisado.

Contudo, como destacado por Koza e Rice (1991), a codificação estruturada requer a utilização de operadores de mutação e cruzamento mais sofisticados que, usualmente, não satisfazem o Teorema dos Esquemas de Holland (1975), necessário à convergência do modelo.

Como agravante, na codificação indireta o operador de cruzamento convencional freqüentemente causa a deterioração das características da RNA.

Wanrooij (1994) comenta que devido à dificuldade de elaboração de operadores de cruzamento mais complexos que respeitem ao Teorema dos Esquemas, diversas abordagens de AGs evitam a utilização deste operador.

No caso da codificação direta, se buscarmos otimizar apenas os pesos sinápticos da RNA (considerando constantes as demais características da RNA tais como a topologia e as taxas de aprendizado), o processo de treinamento de uma RNA otimizada por AGs segue um processo similar ao processo utilizado para as RNAs convencionais.

Por exemplo, adaptando o processo definido por Haykin (1999) para o treinamento seqüencial de uma RNA através do algoritmo de back-propagation, identificamos as seguintes etapas:

- 1.) Inicialização dos pesos sinápticos com valores aleatórios (assumindo que não existam informações prévias disponíveis) segundo uma distribuição normal com média zero e desvio padrão apropriados (i.e., que não inicializem $v_j(0)$ na região de saturação da função de ativação $\varphi(\cdot)$).
- 2.) Introdução do vetor de entrada da época em análise e cálculo do vetor de saída $y_j(n)$ da RNA. Nesta etapa, as saídas $y_j(n)$ são atualizadas sequencialmente, da camada de entrada para a camada de saída da RNA, enquanto os pesos sinápticos $w_{ji}(n)$ permanecem inalterados.
- 3.) Cálculo dos ajustes $\Delta w_{ji}(n)$ segundo uma função de aptidão orientada pelas equações 2.2.4.1.14, 2.2.4.1.17 e 2.2.4.1.20 que realiza a busca estocástica, baseada no cruzamento e na mutação binária dos pesos sinápticos, da população quase-ótima.

- 4.) Nesta etapa, os pesos sinápticos $w_{ji}(n)$ também são atualizados seqüencialmente, da camada de saída para a camada de entrada da RNA, enquanto as saídas $y_j(n)$ dos neurônios da RNA permanecem inalteradas.
- 5.) Repetição das etapas 2 e 3 até que os critérios de parada (precisão, quantidade de iterações, redução da capacidade de generalização, etc.) sejam atendidos ou até que todos os vetores de entrada de todas as épocas disponíveis sejam processados.

Finalmente, quando a otimização dos pesos sinápticos pelo AG não apresentar melhorias significativas após um determinado número de iterações (i.e., quando uma solução quase-ótima for identificada pelo AG), substituímos o processo estocástico de seleção genética pelo mecanismo de otimização de Levenberg-Marquardt para acelerar o processo de convergência.

3 METODOLOGIA

Os contratos de opção da BM&F de compra ou venda sobre a taxa de câmbio de reais por dólar comercial (usualmente denominados contratos de opção de dólar à vista) são contratos de opção do tipo europeu cujo ativo-base é a taxa de câmbio de reais por dólar para entrega pronta (PTAX800 – Opção 5).

Inicialmente, utilizamos todas as informações disponíveis sobre os contratos de opção de compra de dólar à vista. Contudo, como a liquidez destes instrumentos no mercado local é limitada, nem todas as informações disponíveis agregaram conhecimento as diferentes RNAs analisadas.

Por exemplo, as informações das opções com vencimentos mais longos (i.e., com vencimentos superiores a um mês) foram consideradas ruído por todas as RNAs treinadas. Nestes casos, quando mitigamos o efeito de overfitting, as RNAs não foram capazes de inferir o prêmio das opções com vencimentos de prazos maiores como veremos no capítulo seguinte.

Apenas os contratos de opção de compra com prazos de vencimento menores apresentam um volume de negócios diário suficiente para a precificação pelo modelo de RNAs, pois estas requerem amostras de tamanho razoável para treinamento e adaptação das suas sinapses.

Dado a restrição de liquidez, para cada preço de exercício, concentramos nossa análise apenas sobre os primeiros vencimentos disponíveis (que, devido à padronização pela BM&F, ocorre sempre no primeiro dia útil do mês subsequente).

Adicionalmente, para mitigar o risco de outliers, consideramos apenas as opções que apresentaram mais de um negócio diário e cuja volatilidade implícita, calculada pela equação de Garman-Kohlhagen, foi inferior a 200%a.a.

Provavelmente, para amostras maiores e para uma RNA com boa capacidade de generalização (i.e., onde o efeito de overfitting seja mitigado), esses limites mínimo de negociações diárias e máximo de volatilidade não precisariam ser adotados pois a RNA seria capaz de identificá-los como distorções.

Buscando analisar um cenário macroeconômico estável, sem choques extremos (como, por exemplo, o Plano Real e a desvalorização cambial), consideramos somente os negócios realizados diariamente entre 04 de janeiro de 1999 e 30 de novembro de 2006. Devido à ausência de informações mais detalhadas, empregamos apenas as cotações e taxas médias dos instrumentos divulgados neste período.

Em relação às análises, discutiremos a implementação dos diferentes modelos de precificação propostos, as características das diferentes RNAs avaliadas e o processo de comparação do custo do delta hedging de cada abordagem utilizados na análise da eficiência da RNA otimizada por AGs.

No caso do modelo de GK, observando as equações 2.2.1.14, 2.2.1.16 e 2.2.1.17, notamos que são necessários os seguintes parâmetros:

- A cotação S do dólar na data da negociação;
- O preço de exercício K ;
- O prazo T até o vencimento;
- A taxa de juros doméstica livre de risco r (contínua e constante) até o vencimento;
- A taxa do cupom cambial q (contínua e constante) até o vencimento;
- A volatilidade do dólar σ .

Adotamos como taxa de juros doméstica livre de risco, a taxa do contrato futuro de taxa média de depósitos interfinanceiros de um dia (usualmente

denominados contratos futuros de DI), cujo primeiro vencimento ocorre na mesma data do vencimento do contrato de opção de compra do dólar comercial.

Esses contratos futuros de DI utilizam base de 252 dias úteis e taxas compostas anualizadas, que apropriadamente convertemos para taxas contínuas efetivas.

Como taxa do cupom cambial, adotamos a taxa do contrato futuro de cupom cambial (usualmente denominados contratos futuros de DDI), cujo primeiro vencimento ocorre também na mesma data dos vencimentos dos contratos de opção de compra do dólar comercial e de futuro de DI. As taxas destes contratos foram convertidas da base linear como 360 dias corridos para taxas contínuas efetivas.

Como volatilidade do dólar, adotamos a volatilidade implícita anualizada encontrada nos contratos de opção de compra do dólar comercial que calculamos utilizando a função *blsimpv* do Financial toolbox do MATLAB, uma vez conhecidos os demais parâmetros das equações 2.2.1.6, 2.2.1.7 e 2.2.1.8.

No caso do prazo até o vencimento, consideramos apenas os dias úteis (adotando um ano base de 252 dias úteis) entre a data de negociação e a data de vencimento da opção. Para consistência com a taxa de juros e com a volatilidade, os prazos foram calculados em anos.

Analisamos também o modelo de Black (onde substituímos S por F_0 e adotamos $r = q$). Contudo, devido à similaridade dos resultados com o modelo de GK, decidimos considerar apenas este último nas comparações.

No caso dos modelos Lattice, adotamos as Árvores Trinomiais pela ausência do erro de truncagem, detalhada por Figlewski e Gao (1997). Para a construção das árvores trinomiais de precificação, utilizamos as equações 2.2.2.25, 2.2.2.26, 2.2.2.27 e 2.2.2.35 e os mesmos parâmetros (incluindo a volatilidade

implícita) utilizados para o modelo de BS. Todos os cálculos utilizaram $N = 50$ (i.e., 50 passos).

No caso das simulações de Monte Carlo, adotamos o modelo com redução de variância utilizando uma variável antitética (vinculada ao comportamento estocástico log-normal do preço S). Para a implementação do modelo de MC, utilizamos as equações 2.2.3.1 e 2.2.3.2 e os mesmos parâmetros utilizados no modelo de BS. Todos os cálculos utilizaram $M = 30000$ (i.e., 30.000 simulações) e N variável, segundo o prazo, em dias úteis, entre a data de negociação e o vencimento da respectiva opção (i.e., de 1 a 21 passos).

Finalmente, no caso dos modelos de RNA otimizadas por AG, foram avaliadas RNAs com diferentes topologias, números de neurônios por camada e funções de ativação usualmente considerando os mesmos parâmetros utilizados pelo modelo de GK.

A tabela 3.1 apresenta as principais arquiteturas analisadas.

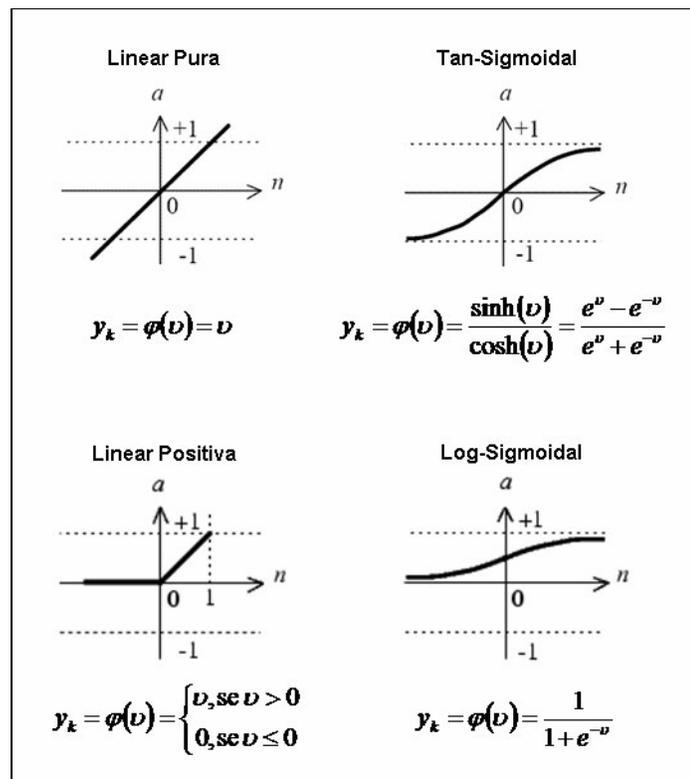
Tabela 3.1: Resumo das Configurações de RNA com AG Analisadas

Configuração da Topologia	Recorrência	Informações de Entrada
10-5-1	Não	S, K, T, r, q, σ
6-3-1	Não	S, K, T, r, q, σ
5-2-1	Não	S, K, T, r, q, σ
5-2-1	Sim	S, K, T, r, q, σ
12-4-1	Não	S, K, T, r, q, σ e preços de commodities
7-4-1	Não	S, K, T, r, q, σ e preços de commodities
7-1	Não	S, K, T, r, q, σ
5-1	Sim	S, K, T, r, q, σ

Detalhando melhor as configurações, comecemos pela configuração 10-5-1 que representa um MLP tradicional (i.e., uma RNA com arquitetura acíclica onde todos os neurônios ou entradas da camada anterior estão conectados a todos os neurônios da camada seguinte) com seis parâmetros de entrada (os mesmos utilizados nos modelos anteriores), dez neurônios na primeira camada, cinco neurônios na segunda e um neurônio na última camada.

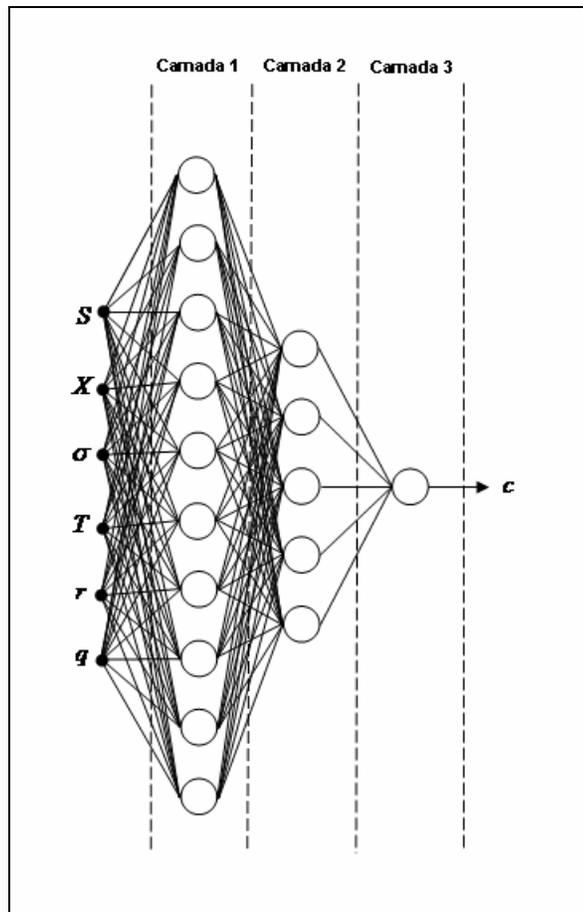
Utilizamos várias combinações de diferentes funções de ativação, exibidas na figura 3.2, nos neurônios da RNA. Entretanto, como veremos no capítulo de Análise, esse modelo não demonstrou bom desempenho, possivelmente devido à quantidade limitada de informações disponíveis para o treinamento dos muitos pesos sinápticos dessa RNA (pois como discutiremos posteriormente, neste processo utilizamos apenas uma parte da já reduzida amostra de informações enquanto empregamos a parte restante nos processos de teste e validação).

Figura 3.1: Funções de Ativação Utilizadas



A figura 3.2 apresenta a topologia da RNA com configuração 10-5-1.

Figura 3.2: Topologia da RNA com Configuração 10-5-1



Prosseguindo, a configuração 6-3-1 apresenta também um MLP similar a RNA com configuração 10-5-1 mas com uma quantidade menor de neurônios. Mesmo com a redução do efeito de overfitting, veremos posteriormente que o desempenho desta configuração também foi insatisfatório.

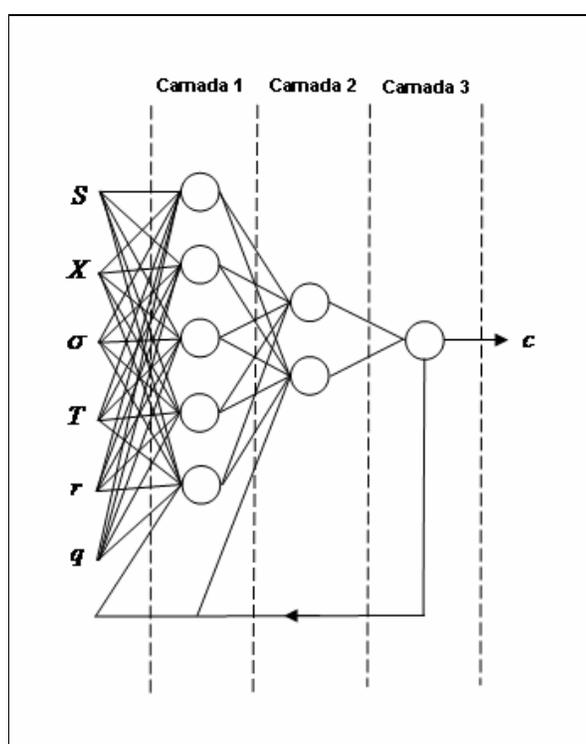
A configuração 5-2-1 com recorrência, baseada na configuração 5-2-1 acíclica que comentaremos posteriormente, não foi capaz de replicar (ou superar) os bons resultados apresentados pela configuração 5-2-1 sem recorrência.

Este desempenho se deve a deterioração causada pelo operador de cruzamento binário durante a otimização dos pesos sinápticos. Para evitar essa

degeneração dos pesos sinápticos, identificamos que seriam necessárias regras adicionais (capazes de proporcionar ao processo a habilidade de contornar o comportamento multimodal e deceptiva das RNAs, que comentaremos posteriormente) para direcionar a evolução genética dos pesos sinápticos.

A figura 3.3 apresenta a topologia da RNA com configuração 5-2-1 com recorrência.

Figura 3.3: Topologia da RNA com Configuração 5-2-1 com Recorrência



As configurações 12-4-1 e 7-4-1 são variações das redes acíclicas discutidas anteriormente que utilizam informações de entrada adicionais, relacionadas aos preços dos contratos futuros de derivativos agropecuários, e buscam avaliar se estes contratos agregam informações relevantes (tais como impactos futuros na balança comercial) ao processo de precificação dos contratos de opções de dólar à vista.

Nestas configurações, utilizamos os seguintes contratos futuros de derivativos agropecuários:

- ✓ Açúcar Cristal Especial
- ✓ Álcool Anidro Carburante
- ✓ Algodão
- ✓ Bezerro
- ✓ Boi Gordo
- ✓ Café Arábica

Contudo, como os vencimentos destes contratos de derivativos agrícolas não são compatíveis com o vencimento dos contratos de opções de dólar à vista, calculamos as taxas implícitas nestes contratos agrícolas ajustando-as pró-rata com base nos dias úteis até o vencimento do respectivo contrato de opção de dólar procurado.

Infelizmente, como veremos no capítulo de Análise, essas informações não proporcionaram nenhuma informação relevante para a precificação dos contratos de opção de dólar.

As figuras 3.4 e 3.5 apresentam as topologias das RNAs com configurações 12-4-1 e 7-4-1.

Figura 3.4: Topologia da RNA com Configuração 12-4-1

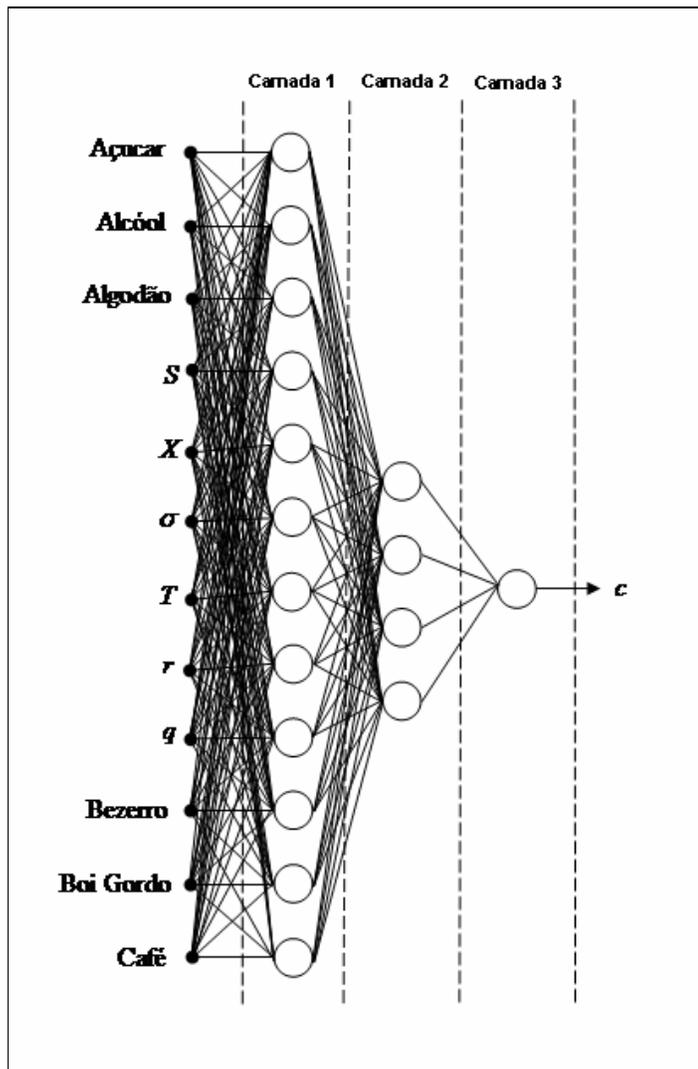
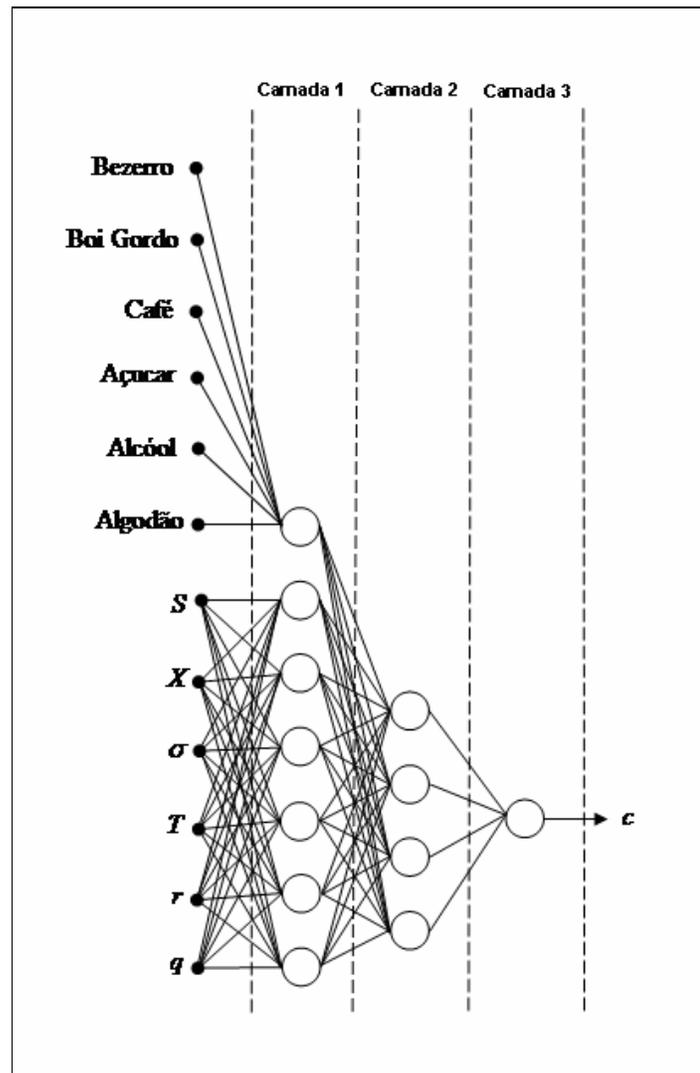


Figura 3.5: Topologia da RNA com Configuração 7-4-1

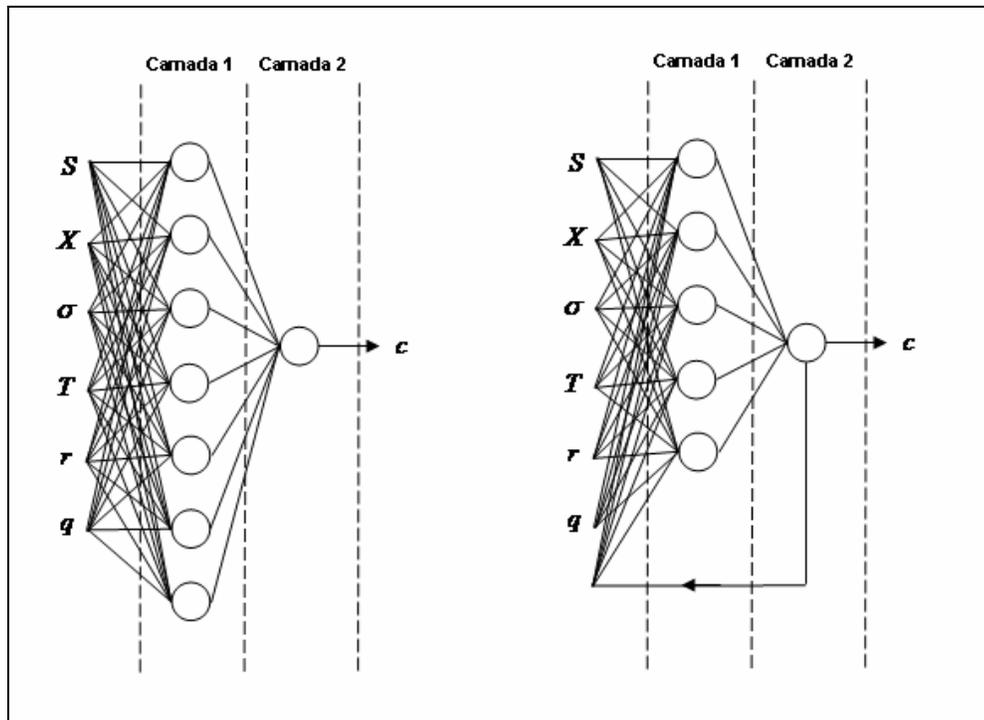


As configurações 7-1 e 5-1 com recorrência são arquiteturas mais simples que não foram capazes de captar adequadamente o conhecimento associado ao comportamento dos preços dos contratos de opção de dólar.

Este comportamento era esperado pois como discutimos anteriormente, segundo Cybenko (1988), são necessárias pelo menos duas camadas intermediárias para permitir que funções matemáticas mais complexas possam ser aproximadas por uma RNA.

A figura 3.6 apresenta as topologias das RNAs com configurações 7-1 e 5-1 com recorrência.

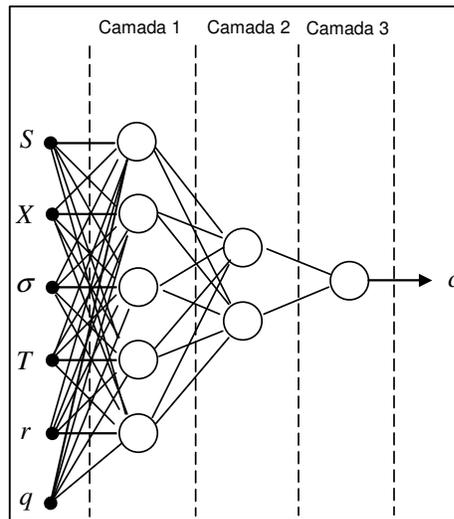
Figura 3.6: Topologias das RNAs com Configurações 7-1 e 5-1 com Recorrência



É importante notarmos que todos os modelos com recorrência analisados, apesar de teoricamente mais qualificados, segundo Haykin (1999), para capturar características de interdependência temporal (como, por exemplo, a heterocedasticidade da volatilidade) apresentaram instabilidade quando otimizados por AGs devido ao operador de cruzamento.

Finalmente, a figura 3.7 apresenta a topologia da RNA com configuração 5-2-1 sem recorrência, responsável pelo melhor desempenho entre as configurações testadas.

Figura 3.7: Topologia com Melhor Desempenho



Detalhando a configuração da RNA apresentada na figura 3.1, encontramos na primeira camada cinco neurônios completamente conectados aos parâmetros de entrada. Cada neurônio utiliza um combinador linear (i.e., um somatório) dos parâmetros de entrada e uma função de ativação tangente hiperbólica.

Analogamente, encontramos na segunda camada dois neurônios completamente conectados às saídas dos neurônios da primeira camada. Cada neurônio desta camada utiliza um combinador linear de suas entradas. Contudo, um dos neurônios utiliza uma função de ativação tangente hiperbólica e outro, uma função linear pura (i.e., uma função de primeiro grau onde $y = x, \forall x \in \mathbb{R}$).

Finalmente, na terceira e última camada notamos que as entradas do único neurônio presente estão conectadas as duas saídas dos neurônios da segunda camada. Este neurônio utiliza um combinador linear para as suas entradas, mas emprega uma função de ativação linear positiva pura (i.e., uma função de primeiro grau onde $y = x, \forall x > 0$ e $y = 0, \forall x \leq 0$) que assegura a inexistência de prêmios negativos.

A função de treinamento da RNA, responsável pelo ajuste dos pesos sinápticos e dos vieses, foi substituída por uma função híbrida associada à função de aptidão do AG.

A função híbrida inicialmente utiliza os mecanismos estocásticos usuais do AG para otimizar os parâmetros da RNA enquanto melhorias relevantes forem identificadas nestes parâmetros.

Contudo, quando ela detecta 20 iterações sucessivas sem variações significativas (i.e., inferiores a 2% do valor absoluto) dos pesos sinápticos e dos vieses, os AGs são substituídos pelo algoritmo de Levenberg-Marquardt para agilizar a convergência para a solução ótima.

A função de performance, responsável pela medição Erro Quadrático Médio (EQM) da RNA que serve como referência à função de treinamento, foi modificada para utilizar o mecanismo de regularização bayesiana, responsável pela minimização do overfitting.

O parâmetro épocas da RNA, responsável pelo número de vezes que o conjunto de informações de treinamento é reprocessado pela função de treinamento, foi ajustado para 100.

Os critérios de parada parcial, utilizados em cada iteração de ajuste dos parâmetros da RNA pelo AG, foram definidos como:

- ✓ O número máximo de 150 gerações ou;
- ✓ O tempo máximo para otimização de 30 segundos ou;
- ✓ A precisão mínima inferior a $1,0E-3$, determinado pela função de performance ou;

Os critérios de parada total do treinamento da RNA foram configurados para observar:

- ✓ A precisão mínima inferior a $1,0E-3$, segundo a função de performance ou;
- ✓ A deterioração da capacidade de diversificação da RNA presente caso a função de performance já tenha alcançado uma precisão mínima de $1,0E-2$ e caso sejam observados pelo menos 15 resultados subseqüentes e contínuos com menor precisão.

Para possibilitar a validação dos critérios de parada e a avaliação não favorecida do modelo de RNA otimizado por AGs, o conjunto de informações disponível composto por $N = 5720$ vetores contendo os preços do ativo base, os preços de exercício, as volatilidades, as taxas de juros e os prazos até o vencimento foi distribuído aleatoriamente em três grupos:

- ✓ Um para treinamento da RNA contendo 60% da amostra;
- ✓ Um para validação da RNA contendo 10% da amostra e;
- ✓ Um para análise da eficiência da RNA contendo os 30% restantes da amostra.

Que foram ordenados sequencialmente segundo a data de vencimento, o preço de vencimento e a data de negociação.

O conjunto de informações de análise, composto por informações de aproximadamente 1700 dias de negociação, proporcionou informações para a análise de 155 operações de delta hedging, calculadas para cada um dos modelo propostos.

Para cada operação de delta hedging, consideramos todos os dias úteis compreendidos entre a data da primeira negociação e a data de vencimento de um dado par “data de vencimento / preço de exercício”.

Para cada dia útil dentro deste período, utilizamos as informações mais recentes disponíveis (preço do ativo base, taxa de juros e volatilidade) para o cálculo do delta da opção, replicando a última volatilidade implícita disponível caso não houvesse negociação no dia útil em análise.

No caso do modelo de BS, o delta utilizado nas operações de delta hedging foi obtido pela função *b/sdelta* do Financial toolbox do MATLAB. Nos demais modelos, o delta foi calculado analiticamente, através da equação:

$$\Delta \approx \frac{\Delta c}{\Delta S} = \frac{c(S + \Delta S, X, r, T, \sigma) - c(S, X, r, T, \sigma)}{\Delta S} \quad (3.1)$$

Onde adotamos $\Delta S = 1,0E - 4$.

Finalmente, comparamos os valores do custo das operações de delta hedging buscando confirmar se o modelo de RNA otimizado por AGs apresenta custos inferiores aos dos demais modelos propostos.

4 ANÁLISES

O primeiro passo da análise compreende a avaliação da capacidade dos modelos de RNAs com AGs de analisar as informações do mercado disponíveis *a priori* (como, por exemplo, o preço do ativo base, a taxa de juros e a volatilidade esperada) e precificar coerentemente (i.e., com significância estatística) as opções utilizando como referência os prêmios observados *a posteriori* no mercado.

Podemos proceder com esta avaliação realizando uma regressão linear pelo método de Mínimos Quadrados Ordinários para confirmar se podemos ou não rejeitar a hipótese nula de que o preço das opções encontrado pelas RNAs (grupo de análise) é similar ao preço negociado no mercado.

$$c_{merc} = \alpha + \beta \cdot c_{RNA} \text{ onde } H_0 : \begin{cases} \alpha = 0 \\ \beta = 1 \end{cases} \quad (4.1)$$

Inicialmente, realizamos as análises das diferentes RNAs discutidas no capítulo anterior utilizando todo o conjunto de informações disponíveis (i.e., todas as informações sobre contratos de opções de compra de dólar à vista disponíveis).

Como sugerem as regressões a seguir, observamos que apenas a configuração 5-2-1 conseguiu precificar com alguma precisão os contratos de opções de dólar.

Figura 4.1: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com
Configuração 10-5-1

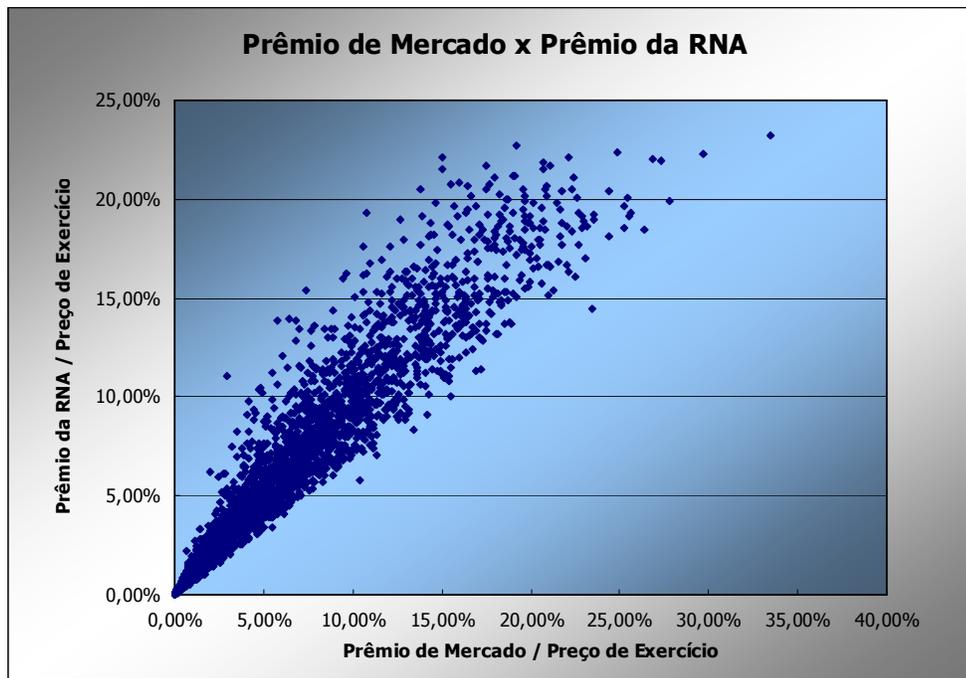


Figura 4.2: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com
Configuração 6-3-1

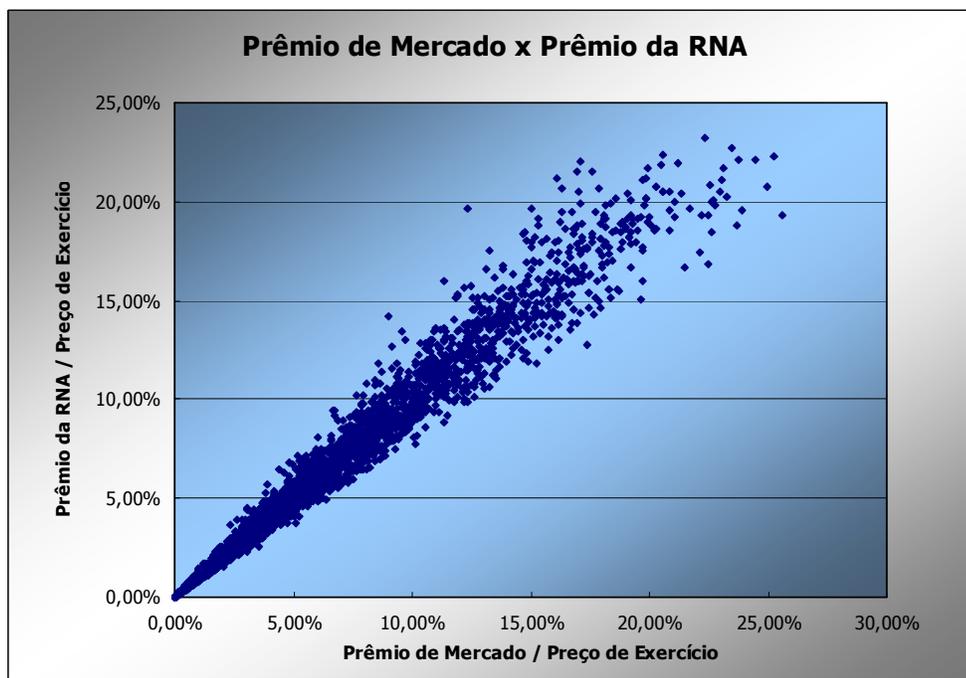


Figura 4.3: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 5-2-1 com Recorrência

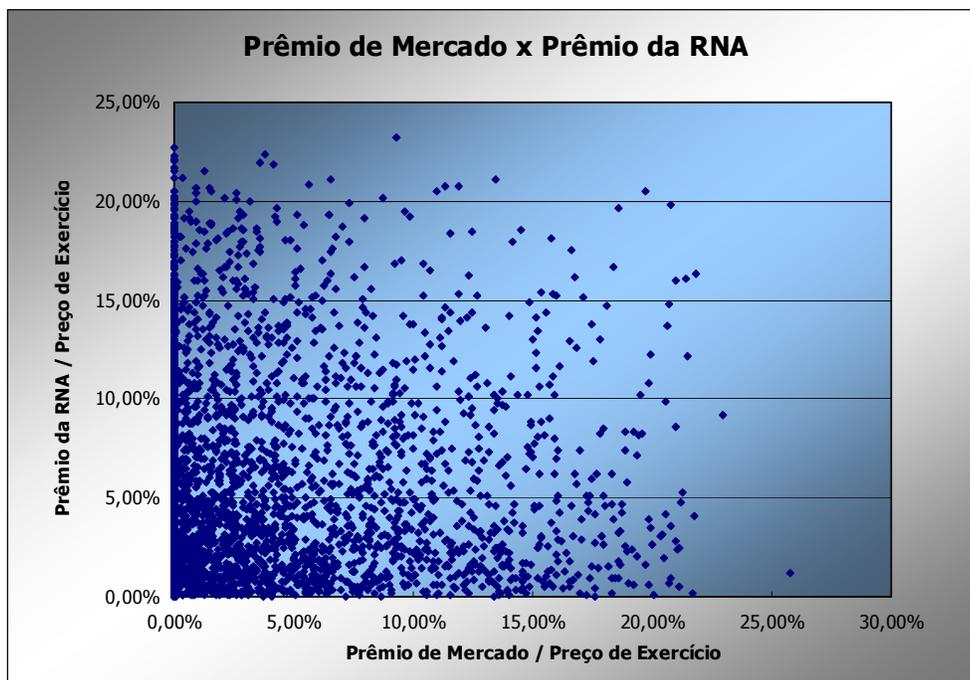


Figura 4.4: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 12-4-1

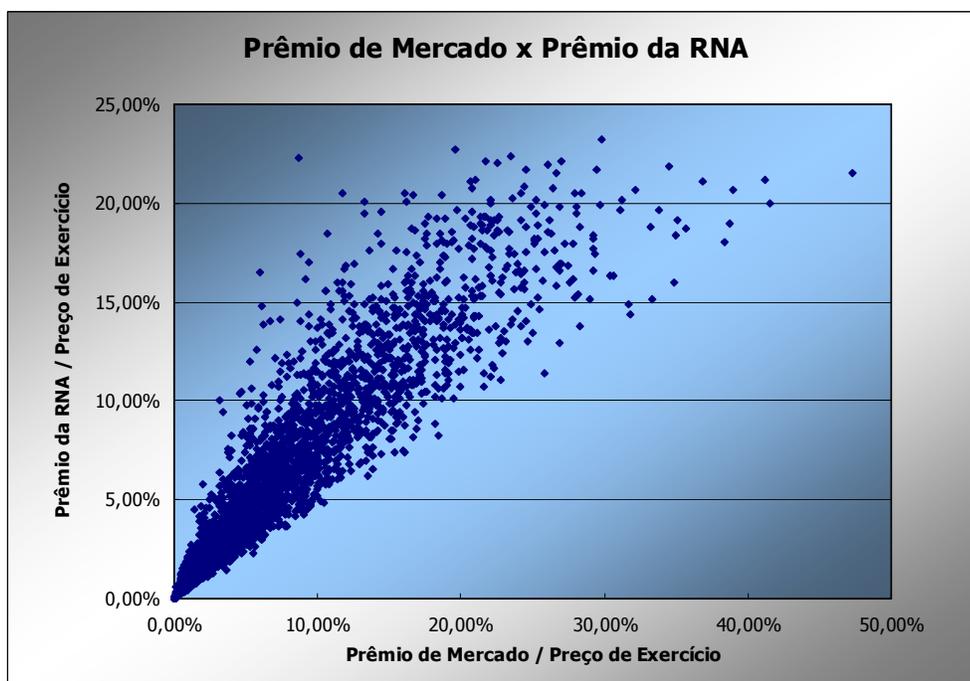


Figura 4.5: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com
Configuração 7-1

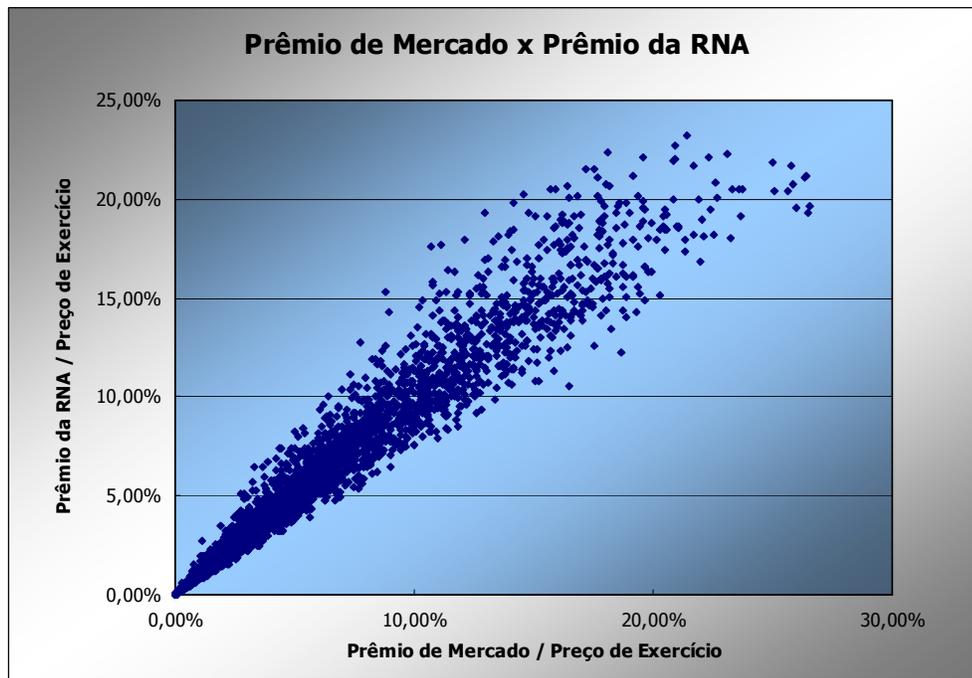
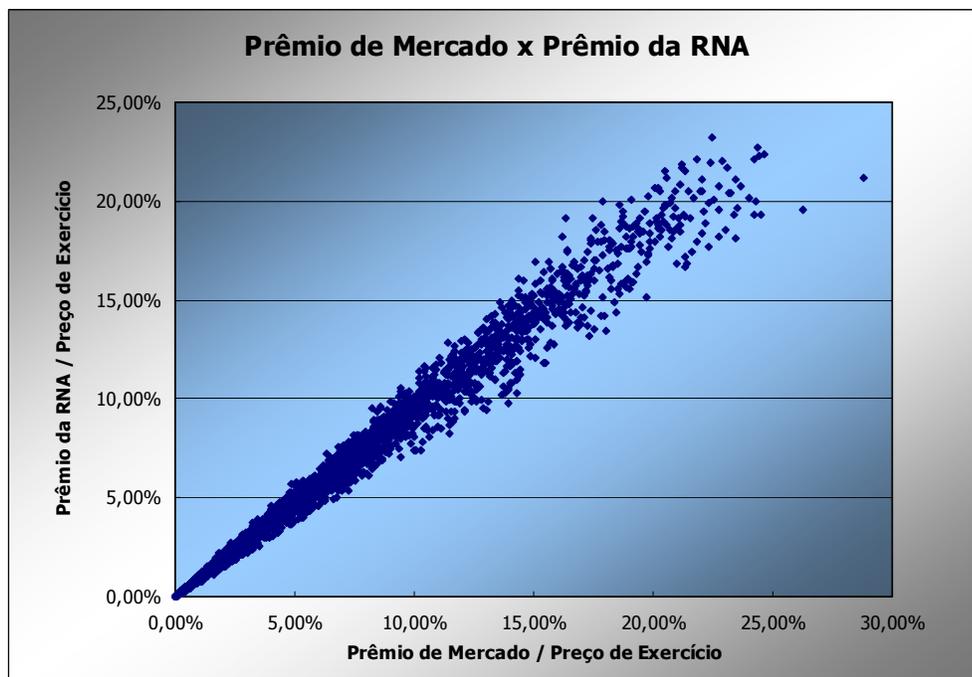


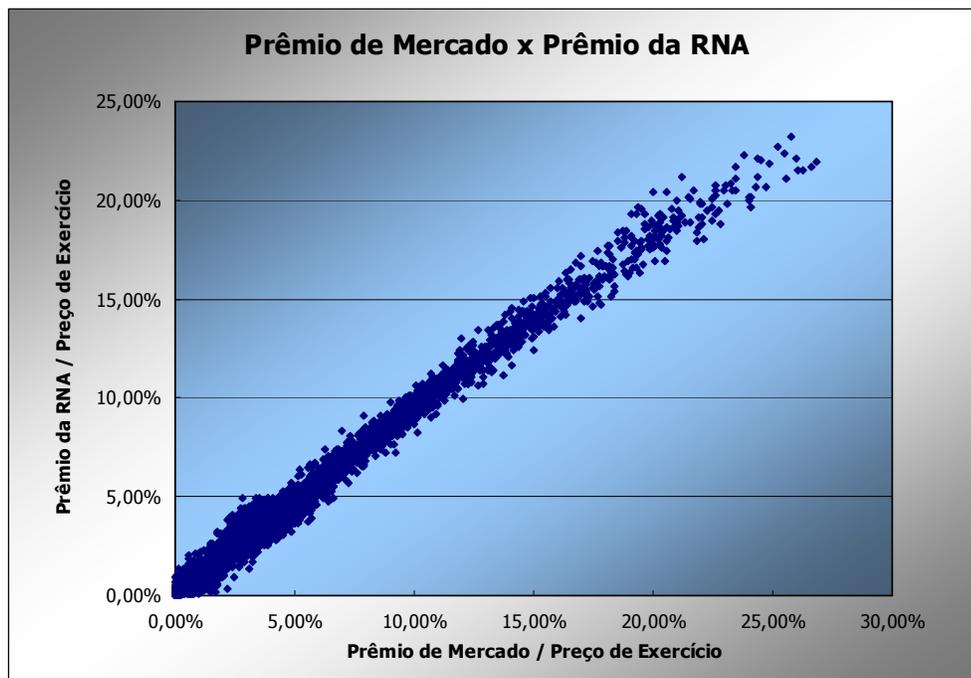
Figura 4.6: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com
Configuração 5-2-1



Contudo, observamos que mesmo esta configuração não apresentou bom desempenho na precificação de opções com vencimentos mais longos. Para identificar as causas deste comportamento, realizamos novamente a regressão da RNA com configuração 5-2-1 removendo o mecanismo de regularização bayesiana utilizado para minimizar o efeito de overfitting.

Pela figura 4.7, notamos que esta modificação proporcionou uma melhora na precificação das opções com vencimentos mais longos mas causou um aumento significativo do overfitting das informações de treinamento (i.e., uma redução relevante na capacidade de precificação da RNA).

Figura 4.7: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com Configuração 5-2-1 sem Regularização Bayesiana



Podemos supor, portanto, que devido à significativa escassez de informações disponíveis para as opções de vencimento mais longo, o processo de treinamento com regularização bayesiana adequou os pesos sinápticos

responsáveis pela precificação destes contratos para que os mesmos não comprometessem a capacidade de generalização da RNA.

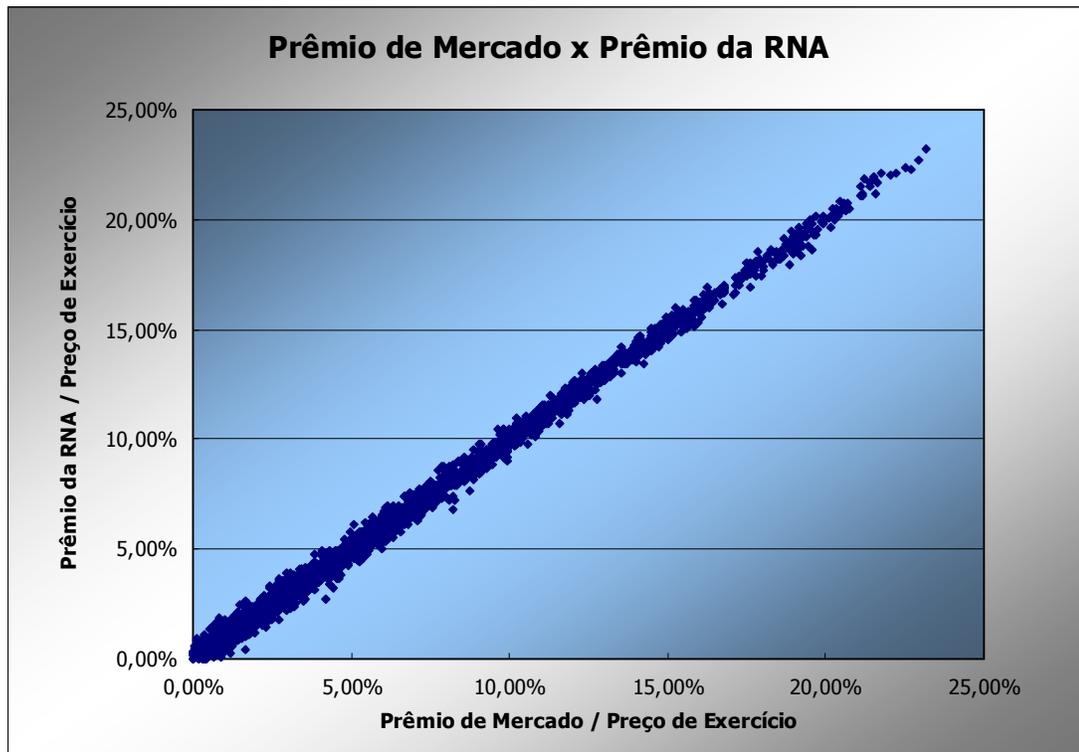
Prosseguindo com as análises, restringimos o conjunto de informações e utilizamos apenas os contratos de opção com vencimento no primeiro dia útil do mês subsequente a negociação.

Contudo, ainda obtivemos convergência limitada no treinamento das RNAs devido às informações oriundas dos dias com poucas negociações que, em diversos casos, incluíram ruídos significativos nas informações de treinamento utilizadas.

Nestes casos, como explicamos previamente no capítulo de Metodologia, optamos por excluir estas informações adotando um limite mínimo de negócios diários e um limite máximo de volatilidade implícita (segundo o modelo de GK) para determinar as informações diárias que deveriam ser utilizadas no treinamento das RNAs.

A figura a seguir apresenta a regressão da RNA com configuração 5-2-1 sem recorrência considerando as informações filtradas.

Figura 4.8: Comparação do Preço de Mercado da Opção com o Preço calculado pela RNA com configuração 5-2-1



Observamos que a figura 4.8 indica uma boa convergência da RNA com configuração 5-2-1 que podemos confirmar utilizando a função *regress* do MATLAB

$$\begin{aligned}
 \alpha &= 5,18e - 5, p_{\alpha} = 0 \\
 \beta &= 0,999, p_{\beta} = 9,48e - 006 \\
 R^2 &= 0,9963
 \end{aligned}
 \tag{4.2}$$

Portanto, nesta configuração, não podemos rejeitar a hipótese nula que sugere que o modelo de RNAs com AGs proporciona preços próximos àqueles praticados pelo mercado efetivamente.

Este resultado é particularmente interessante porque sugere que a utilização dos AGs no processo de treinamento das RNAs elimina ou, pelo menos, reduz significativamente as limitações decorrentes de eventuais mínimos locais que

são responsáveis pelas precificações exacerbadamente incorretas nos modelos convencionais de RNAs.

Outro aspecto relevante da configuração 5-2-1 corresponde ao desempenho similar apresentado no tratamento de opções de compra de dólar fora, dentro ou no dinheiro (i.e., opções cuja cotação do dólar está abaixo, acima ou próxima ao seu preço de exercício) o que sugere que o modelo capta adequadamente o comportamento destas opções nestas situações.

Entretanto, devemos lembrar que mesmo com os resultados observados acima os modelos de RNAs não possibilitam a construção analítica dos intervalos de confiança e que, portanto, não podemos determinar, *a priori*, o erro decorrente da precificação por estes modelos.

Em relação aos demais modelos, podemos destacar que:

- ✓ A configuração 10-5-1 não apresentou bom desempenho possivelmente devido à quantidade elevada de pesos sinápticos presentes nesta configuração. Mesmo a combinação de diferentes funções de ativação não foi suficiente para compensar a quantidade limitada de informações disponíveis para treinamento da RNA.
- ✓ A configuração 6-3-1, com limitações semelhantes e desempenho pouco superior à configuração 10-5-1, ofereceu ainda resultados insatisfatórios (i.e., com elevada ocorrência de preços demasiadamente incorretos).
- ✓ As configurações 12-4-1 e 7-4-1, que incluíram como informações de entrada adicionais no treinamento das RNAs os contratos de derivativos agrícolas, demandaram um período muito mais longo de treinamento sem oferecer um bom desempenho (inferior a quase todas as outras configurações). Este comportamento sugere que os contratos futuros

destas commodities não contêm informações relevantes para a precificação dos contratos de opção de dólar.

Entretanto, é importante ressaltarmos que aspectos como a abordagem adotada para o ajuste das taxas implícitas destes contratos agrícolas e a ausência de um tratamento apropriado para os fatores de produção e consumo associados aos ativos base destes instrumentos pode ter prejudicado a identificação de quaisquer informações úteis presentes nestes contratos.

- ✓ A configuração 5-2-1 com recorrência apresentou péssimo desempenho decorrente da incompatibilidade do operador de cruzamento direto (i.e., binário) com a arquitetura recorrente da RNA.

Neste caso, identificamos que o comportamento multimodal (i.e., diferentes arquiteturas com resultados semelhantes) e deceptivo (i.e., arquiteturas semelhantes com resultados drasticamente distintos) das RNAs inviabiliza a utilização do operador de cruzamento tradicional.

- ✓ As configurações 7-1 sem recorrência e 5-1 com recorrência, devido à menor quantidade de camadas intermediárias, não foram capazes de modelar corretamente o comportamento dos preços dos contratos de opção de dólar. Esse comportamento se deve a função matemática relativamente complexa associada a este comportamento dos preços dessas opções que exige, segundo Cybenko (1988), uma arquitetura mais sofisticada. Adicionalmente, a RNA com configuração 5-1 com recorrência apresentou as mesmas dificuldades observadas previamente com o operador de cruzamento.

Comparadas as diferentes RNAs, adotamos a configuração 5-2-1 como a mais adequada e a utilizamos na etapa seguinte da análise que envolveu a comparação dos custos das operações de hedge dinâmico das opções calculados pelo rebalanceamento dos portfólios segundo os deltas fornecidos pelos diferentes modelos analisados.

A tabela 4.1 apresenta um resumo dos resultados obtidos.

Tabela 4.1: Comparação dos Custos de Delta-Hedging dos Modelos Analisados

	BS	TR	MC	RNA
MÉDIA	0,0269	0,0268	0,0258	0,0266
DESVIO	0,0361	0,0355	0,0418	0,0380
MÁXIMO	0,2017	0,2035	0,2429	0,2044
MÍNIMO	-0,0976	-0,0887	-0,1263	-0,0736

	RNA - BS	RNA - TR	RNA - MC
MÉDIA	-0,0003	-0,0002	0,0008
DESVIO	0,0122	0,0122	0,0262
MÁXIMO	0,0352	0,0354	0,2659
MÍNIMO	-0,0411	-0,0394	-0,1322

Para avaliar estes resultados, realizamos testes de hipótese das diferenças entre as médias encontradas.

Adotando

$$H_0 : \begin{cases} \mu_{RNA} - \mu_{BS} = 0 \\ \mu_{RNA} - \mu_{TR} = 0 \\ \mu_{RNA} - \mu_{MC} = 0 \end{cases} \quad (4.3)$$

Sabemos que

$$z_{RNA-BS} = \frac{(\bar{x}_{RNA} - \bar{x}_{BS}) - (\mu_{RNA} - \mu_{BS})}{\sqrt{\frac{\sigma_{RNA}^2}{N_{RNA}} + \frac{\sigma_{BS}^2}{N_{BS}}}} = \frac{(-0,003) - (0,0)}{\sqrt{\frac{0,0380^2}{155} + \frac{0,0361^2}{155}}} = -0,0840 \quad (4.4)$$

E, analogamente

$$z_{RNA-TR} = \frac{(-0,002) - (0,0)}{\sqrt{\frac{0,0380^2}{155} + \frac{0,0355^2}{155}}} = -0,0486 \quad (4.5)$$

$$z_{RNA-MC} = \frac{(0,008) - (0,0)}{\sqrt{\frac{0,0380^2}{155} + \frac{0,0418^2}{155}}} = 0,2067 \quad (4.6)$$

Onde observamos que para os níveis usuais de significância, os testes de hipótese confirmam que, exceto para o modelo de Simulações de Monte Carlo (cuja precisão é estatisticamente superior), não podemos rejeitar a hipótese nula.

Portanto, não podemos afirmar que o modelo híbrido composto pela RNA e pelo AG proporciona resultados estatisticamente diferentes dos modelos de precificação de BS e de Árvores Trinomiais.

É importante notarmos que a quantidade limitada de operações de delta hedging disponíveis ($N = 155$), decorrente da impossibilidade de utilização das informações utilizadas no treinamento e na validação do modelo de RNAs (que juntas constituem 70% do espaço amostral total), é um fator negativo relevante pois reduz a precisão dos testes estatísticos realizados.

5 CONSIDERAÇÕES FINAIS

Neste trabalho, analisamos a precisão de um modelo simples de RNAs otimizadas por AGs para a precificação de opções de dólar à vista e observamos que o mesmo não apresentou resultados estatisticamente superiores àqueles dos modelos convencionais.

No entanto, a ausência de preços expressivamente distorcidos na precificação de opções pelo modelo de RNAs otimizadas por AGs indicou uma característica positiva desta abordagem que superou as dificuldades dos modelos baseados em RNAs convencionais no processamento de superfícies mais complexas (i.e., com mínimos locais).

Devemos ressaltar que o principal fator restritivo da análise deste trabalho foi o reduzido número de informações disponíveis (i.e., as informações do grupo de análise) para comparação dos modelos. Ainda que a liquidez do mercado doméstico de opções esteja aumentando gradualmente nos últimos anos com a consolidação e evolução dos fundamentos macroeconômicos, a utilização destes instrumentos derivativos ainda é limitada para permitir o uso eficiente de algumas abordagens não-paramétricas como as RNAs.

Adicionalmente, devido a complexidade envolvida na elaboração de modelos híbridos de RNAs e AGs, um tema relativamente novo na literatura internacional segundo Chen (2002), diversas alternativas mais sofisticadas não foram abordadas neste trabalho e podem ser posteriormente exploradas.

Uma destas alternativas para o aperfeiçoamento do modelo híbrido discutido neste trabalho é a substituição das RNAs acíclicas por RNAs recorrentes, como proposto na RNA de configuração 5-2-1 com recorrência, capazes de

armazenar e construir padrões temporais, que, segundo Haykin (1999), são mais adequadas para estimar a volatilidade implícita do ativo-base.

Como mencionamos, neste trabalho, identificamos que a incompatibilidade das RNAs recorrentes com os AGs está relacionada principalmente à construção e implementação do operador de cruzamento que, mesmo na codificação direta, precisaria ser modificada para garantir a estabilidade da convergência. Infelizmente, ainda existem poucos estudos detalhados sobre este tipo de sistema híbrido.

Outra alternativa para a extensão deste trabalho, bem mais explorada que a sugestão anterior, é a utilização da codificação indireta. Os trabalhos de Montana (1992) e Koehn (1994) são excelentes referências sobre as vantagens e dificuldades adicionais que a representação indireta do fenótipo proporciona.

A terceira alternativa envolve a avaliação de outros parâmetros de entrada como propusemos, por exemplo, nas RNAs de configurações 12-4-1 e 7-4-1 onde buscamos identificar novas variáveis relevantes (i.e., que contenham informações adicionais àquelas contidas no conjunto de parâmetros atualmente utilizado) ao processo de precificação destes instrumentos. Freitas (2001) apresenta algumas variações interessantes dos parâmetros de entrada.

Finalmente, é importante observamos que, conforme demonstrado por Yao (1999), os modelos de RNAs impõe poucas restrições ou hipóteses ao processo de precificação. Conseqüentemente, havendo informações suficientes para o treinamento, validação e testes adequados do modelo, podemos utilizar esta abordagem para modelar instrumentos derivativos mais complexos.

6 REFERÊNCIAS BIBLIOGRÁFICAS

- [1] **BLACK, F.** The Pricing of Commodity Contracts. *Journal of Financial Economics*, n.3, p.167-179, mar., 1976.
- [2] **BLACK, F.; DERMAN, E.; TOY, W.** A One-Factor Model of Interest Rates and Its Application to Treasury Bond Options. *Financial Analysts Journal*, n.46, p.33-39, jan./feb., 1990.
- [3] **BLACK, F.; SCHOLES, M.** The Pricing of Options and Corporate Liabilities. *Journal of Political Economy*, n.81, p.637-654, mai./jun., 1973.
- [4] **BOYLE, P. P.** Options: A Monte Carlo Approach. *Journal of Financial Economics*, n.4, p.323-338, mai., 1977.
- [5] _____. Option Valuation Using a Three Jump Process. *International Options Journal*, n.3, n.7-12, 1986.
- [6] _____. A Lattice Framework for Option Pricing with Two State Variables. *Journal of Financial and Quantitative Analysis*, n.23, p.1-12, mar., 1988.
- [7] **BOYLE, P. P.; EVNINE, J.; GIBBS, S.** Numerical Evaluations of Multivariate Contingent Claims. *Review of Financial Studies*, n.2, p.241-250, 1989.
- [8] **BOYLE, P. P.; BROADIE, M.; GLASSERMAN, P.** Monte Carlo Methods for Security Pricing. *Journal of Economic Dynamics and Control*, n.21, p.1267-1321, 1997.
- [9] **BOYLE, P. P.; JOY, C.; TAN, K. S.** Quasi-Monte Carlo Methods in Numerical Finance. *Management Science*, v.42, n.6, p.926-938, jun., 1996.
- [10] **BRAGA, A. P.; LUDEMIR, T. B.; CARVALHO, A. C. P. L. F.** *Redes Neurais Artificiais*, Rio de Janeiro: Livros Técnicos e Científicos Editora, 2000.

- [11] **BREEN, R.** The Accelerated Binomial Option Pricing Model. *Journal of Financial and Quantitative Analysis*, n.26, p.153-164, 1991.
- [12] **BROADIE, M.; GLASSERMAN, P.; JAIN, G.** Enhanced Monte Carlo Estimates for American Option Prices. *The Journal of Derivatives*, n.5, p.25-44, 1997.
- [13] **BROTHERTON-RATCLIFFE, R.** Monte Carlo Motoring. *Risk*, v.7, n.4, p.53-58, dez., 1994.
- [14] **CHARALAMBOUS, C.** Conjugate Gradient Algorithm for Efficient Training of Artificial Neural Networks, *IEEE Proceedings*, v.139, n.3, p.301-310, jun., 1992.
- [15] **CHEN, S.** Genetic Algorithms and Genetic Programming in Computational Finance, Norwell: Kluwer Academic Publishers, 2002.
- [16] **CHEN, S.; LEE; W.** Option Pricing with Genetic Algorithms: The Case of European Options. *Proceedings of International Conference on Genetic Algorithms*, p.704-711, San Francisco, 1997.
- [17] _____. Option Pricing with Genetic Algorithms: Separating Out-of-the-Money from In-the-Money. *Proceedings of International Conference on Intelligent Processing Systems*, v.2, p.110-115, Beijing, 1997.
- [18] **CLEWLOW, L.; STRICKLAND, C.** Implementing Derivatives Models, New York: John Wiley & Sons Ltd, 1998.
- [19] **COX, J. C.; ROSS, A.; RUBINSTEIN, M.** Option Pricing: A Simplified Approach. *Journal of Financial Economics*, n.7, p.229-263, 1979.
- [20] **DASGUPTA, D.; MCGREGOR, D.** Designing Application Specific Neural Networks using the Structured Genetic Algorithm. *Proceedings of the International Conference on Combinations of Genetic Algorithms and Neural Networks*, p.87-96, Piscataway, 1992.

- [21] **DAVIS, L.** Handbook of Genetic Algorithms, New York: Van Nostrand Reinhold Publisher, 1991.
- [22] **DEFALCO, I.; IAZZETTA, A.; NATALE, P.; TARANTINO, E.** Evolutionary Neural Networks for Nonlinear Dynamics Modeling, Lectures in Computer Science, v.1498, p.593-602, 1998.
- [23] **DEMUTH, H. B.; BEALE, M. H.** Neural Networks Toolbox for Use with Matlab, The Mathworks Inc., 1993.
- [24] **DENNIS, J. E.; SCHNABEL, R. B.** Numerical Methods for Unconstrained Optimization and Non-Linear Equations, New Jersey: Prentice Hall, 1983.
- [25] **DORFFNER, G.** Neural Networks for Time Series Processing. Neural Network World, n.6, p.447-468, 1996.
- [26] **ELMAN, J. L.** Finding Structure in Time. Cognitive Science, n.14, p.179-211, 1990.
- [27] **FERNANDES, M. A.** Precificação e Hedge Dinâmico de Opções de Telebrás utilizando Redes Neurais. Porto Alegre, 2000. Tese (Mestrado em Administração). Escola de Administração, Universidade Federal do Rio Grande do Sul - RS.
- [28] **FREITAS, S. O.** Utilização de um Modelo baseado em Redes Neurais para a Precificação de Opções. Belo Horizonte, 2001. Dissertação (Mestrado em Administração). Faculdade de Ciências Econômicas, Universidade Federal de Minas Gerais - MG.
- [29] **FIGLEWSKI, S.; GAO, B.** The Adaptive Mesh model: a New Approach to Efficient Option Pricing. Journal of Financial Economics, n.53, p.331-351, 1999.
- [30] **GARMAN, M. B.; KOHLHAGEN, S. W.** Foreign Currency Option Values. Journal of International Money and Finance, n.2, p.231-237, dez., 1983.

- [31] **GOLDBERG, D. E.** Genetic Algorithms in Search, Optimization, and Machine Learning, Boston: Kluwer Academic Publishers, 1989.
- [32] **GRUAU, F.** Genetic Synthesis of Modular Neural Networks. Proceedings of the Fifth International Conference on Genetic Algorithms, p.318–325, San Francisco, 1993.
- [33] **GRUAU, F.; WHITLEY, D.; PYEATT, L.** A Comparison between Cellular Encoding and Direct Encoding for Genetic Neural Networks. Proceedings of the First Annual Conference, p.81-89, Stanford, 1996.
- [34] **HAGAN, M. T.; DEMUTH, H. B.; BEALE, M. H.** Neural Network Design, 1^a Edição, Boston: PWS Publishing, 1996.
- [35] **HAUG, E. G.** The Complete Guide to Option Pricing Formulas, New York: McGraw-Hill, 1997.
- [36] **HAYKIN, S.** Neural Networks, 2^a Edição, New Jersey: Prentice Hall, 1999.
- [37] **HOLLAND, J.** Adaptation in Natural and Artificial Systems, Ann Arbor: University of Michigan Press, 1975.
- [38] **HULL, J.** Introduction to Futures and Option Markets, 2^a Edição, New Jersey: Prentice Hall, 1996.
- [39] _____. Options, Futures and Other Derivatives, 5^a Edição, New Jersey: Prentice Hall, 2002.
- [40] **HULL, J.; WHITE, A.** The Pricing of Options on Assets with Stochastic Volatilities. Journal of Finance, n.42, p.281–300, 1987.
- [41] _____. The Use of the Control Variate Technique in Option Pricing. Journal of Financial and Quantitative Analysis, n.23, p.237–251, 1988.
- [42] **JARROW, R.; RUDD, A.** Option Pricing, Homewood: Irwin Publishing, 1983.

- [43] **KAMAKURA, A.** Otimização de Hedge de Derivativos utilizando Algoritmos Genéticos. São Paulo, 2004. Dissertação (Mestrado em Modelagem Matemática em Finanças). Faculdade de Economia e Administração, Universidade de São Paulo - SP.
- [44] **KOEHN, P.** Combining Genetic Algorithms and Neural Networks: The Encoding Problem. Knoxville, 1994. Thesis (Master of Computer Science). University of Tennessee.
- [45] **KOZA, J. R.; RICE, J. P.** Genetic Generation of both the Weights and Architecture for a Neural Network. IEEE International Joint Conference on Neural Networks, p.397-404, Seattle, 1991.
- [46] **LANARI, C. S.; SOUZA, A. A.; DUQUE, J. C.** Desvios em Relação ao Modelo de Black e Scholes: Estudos Relacionados à Volatilidade dos Ativos Subjacentes às Opções. III Encontro de Engenharia de Produção da UFRJ, p.1-14, Rio de Janeiro, 1999.
- [47] **MADAN, D.; MILNE, F.; SHEFRIN, H.** The Multinomial Option Pricing Model and Its Brownian and Poisson Limits. Review of Financial Studies, n.2, p.251-266, 1989.
- [48] **MANDISCHER, M.** Representation and Evolution of Neural Networks. Proceedings of the International Conference on Artificial Neural Networks and Genetic Algorithms, p.643-649, Innsbruck, Austria, 1993.
- [49] **MANIEZZO, V.** Genetic Evolution of the Topology and Weight Distribution of Neural Networks. IEEE Transactions on Neural Networks, v.5, p.39-53, 1994.
- [50] **MARQUARDT, D.** An Algorithm for Least-Squares Estimation of Nonlinear Parameters. Journal of Applied Mathematics, n.11, p.431-441, 1963.

- [51] **MCNELIS, P. D.** Neural Networks in Finance, San Diego: Elsevier Academic Press, 2005.
- [52] **MERTON, R. C.** Theory of Rational Option Pricing. Bell Journal of Economics and Management Science, n.4, p.141-183, 1973.
- [53] **MITCHELL, M.** An Introduction to Genetic Algorithms, Cambridge: MIT Press, 1996.
- [54] **MONTANA, D. J.** A Weighted Probabilistic Neural Network. Advances in Neural Information Processing Systems, n.4, p.1110-1117, 1992.
- [55] **MONTANA, D. J.; DAVIS, L.** Training Feedforward Neural Networks Using Genetic Algorithms. Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, p.762-767, Detroit, 1989.
- [56] **NAWALKHA, S. K.; CHAMBERS, D. R.** The Binomial Model and Risk Neutrality: Some Important Details. The Financial Review, n.30, p.605-615, 1995.
- [57] **PARKINSON, M.** Option Pricing: The American Put. Journal of Business, n.50, p.21-36, 1977.
- [58] **PRUDÊNCIO, R. B. C.** Projeto Híbrido de Redes Neurais. Recife, 2002. Tese (Mestrado em Ciência da Computação). Departamento de Informática, Universidade Federal de Pernambuco.
- [59] **RENDLEMAN, R. J.; BARTTER, B. J.** Two-state Option Pricing. Journal of Finance, n.34, p.1093-1110, dez., 1979.
- [60] **ROCHMAN, R. R.** Análise de Métodos Numéricos para Precificação de Opções. São Paulo, 1998. Tese (Mestrado em Administração de Empresas). Escola de Administração de Empresas de São Paulo, Fundação Getúlio Vargas - São Paulo.

- [61] **RUBINSTEIN, M. E.** Implied Binomial Trees. *Journal of Finance*, n.69, p.771-818, 1994.
- [62] **RUMELHART, D. E.; MCCLELLAND, J. L.** *Parallel Distributed Processing*, v.1, Cambridge: MIT Press, 1986.
- [63] **SCALES, L. E.** *Introduction to Non-Linear Optimization*, New York: Springer-Verlag, 1985.
- [64] **SCHAFFER, J. D.; CARUANA, R. A.; ESHELMAN, L. J.** Using Genetic Search to Exploit the Emerging of Neural Networks. *Proceedings of the Ninth Annual International Conference of the Center for Nonlinear Studies on Self-organizing, Collective, and Cooperative Phenomena in Natural and Artificial Computing Networks on Emergent Computation*, p.244-248, Los Alamos, 1990.
- [65] **SCHAFFER, J. D.; WHITLEY, D.; ESHELMAN, L. J.** Combinations of Genetic Algorithms and Neural Networks: A Survey of the State of the Art. *Proceedings of the IEEE Workshop on Combinations of Genetic Algorithms and Neural Networks*, p.1-37, 1992.
- [66] **STEPNIEWSKI, S. W.; KEANE, A. J.** Topology Design of Feedforward Networks by Genetic Algorithms. *Parallel Problem Solving from Nature*, p.771-780, Berlin: Springer-Verlag, 1996.
- [67] **TIAN, Y.** A Modified Lattice Approach to Option Pricing. *Journal of Futures Markets*, n.13, p.564-577, 1993.
- [68] **TRIGEORGIS, L.** A Log-Transformed Binomial Numerical Analysis Method for Valuing Complex Multi-Option Investments. *Journal of Financial and Quantitative Analysis*, n.26, p.309-326, 1991.
- [69] **VALENÇA, M. J. S.** *Análise e Projeto de Redes Neurais com Abordagem Construtiva para Modelagem de Sistemas Complexos*. Recife, 1999. Tese

(Doutorado em Ciência da Computação). Departamento de Informática, Universidade Federal de Pernambuco.

- [70] **WANROOIJ, E.** Evolving Sequential Neural Networks for Time Series Forecasting. Utrecht, 1994. Thesis (Master in Computer Science). Department of Computer Science, University of Utrecht.
- [71] **WHITLEY, D.; STARKWEATHER, T.; BOGART, C.** Genetic Algorithms and Neural Networks: Optimizing Connections and Connectivity. *Parallel Computing*, v.14, p.347-361, 1990.
- [72] **WHITE, J. A.** Pricing Options with Futures-Style Margining: A Genetic Adaptive Neural Network Approach, New York: Taylor and Francis, 2000.
- [73] **YAO, X.** Evolving Artificial Neural Networks. *Proceedings of the IEEE*, v.87, p.1423-1447, 1999.
- [74] **ZHANG, B. T.; MUHLENBEIN, H.** Evolving Optimal Neural Networks Using Genetic Algorithms with Occam's Razor. *Complex Systems*, n.7, p.199-220, 1993.

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)