

**UNIVERSIDADE DE SÃO PAULO**  
**ESCOLA DE ENGENHARIA DE SÃO CARLOS**

FELIPE TUMENAS MARQUES

**OTIMIZAÇÃO DE CARTEIRAS COM LOTES DE COMPRA E**  
**CUSTOS DE TRANSAÇÃO, UMA ABORDAGEM POR**  
**ALGORITMOS GENÉTICOS**

São Carlos

2007

# **Livros Grátis**

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

Milhares de livros grátis para download.

FELIPE TUMENAS MARQUES

**OTIMIZAÇÃO DE CARTEIRAS COM LOTES DE COMPRA E  
CUSTOS DE TRANSAÇÃO, UMA ABORDAGEM POR  
ALGORITMOS GENÉTICOS**

Dissertação apresentada à Escola de  
Engenharia de São Carlos da Universidade  
de São Paulo para obtenção do Título de  
Mestre em Engenharia de Produção

Área de Concentração: Engenharia de  
Produção

Orientador: Prof. Dr. Marcelo Seido  
Nagano

São Carlos

2007

**AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTES  
TRABALHOS, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO,  
PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.**

**Ficha catalográfica preparada pela Seção de Tratamento  
da Informação do Serviço de Biblioteca – EESC/USP**

M357o Marques, Felipe Tumenas  
Otimização de carteiras com lotes de compra e custos  
de transação : uma abordagem por algoritmos genéticos /  
Felipe Tumenas Marques ; orientador Marcelo Seido Nagano.  
-- São Carlos, 2007.


Dissertação (Mestrado) - Programa de Pós-Graduação e  
Área de Concentração em Engenharia de Produção -- Escola  
de Engenharia de São Carlos da Universidade de São Paulo.


1. Finanças . 2. Otimização de carteiras. 3.  
Markowitz. 4. Algoritmos genéticos. I. Título.


**FOLHA DE JULGAMENTO**

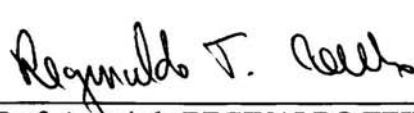
Candidato: Bacharel **FELIPE TUMENAS MARQUES**

Dissertação defendida e julgada em 02/10/2007 perante a Comissão Julgadora:

  
Prof. Dr. **MARCELO SEIDO NAGANO** (Orientador)  
(Escola de Engenharia de São Carlos/USP) **APROVADO**

  
Prof.<sup>a</sup> Associada **DAISY APARECIDA DO NASCIMENTO REBELATTO**  
(Escola de Engenharia de São Carlos/USP) **APROVADO**

  
Prof. Dr. **ALCEU SALLES CAMARGO JÚNIOR**  
(Faculdade de Economia, Administração e Contabilidade de Ribeirão Preto/USP) **APROVADO**

  
Prof. Associado **REGINALDO TEIXEIRA COELHO**  
Coordenador do Programa de Pós-Graduação em  
Engenharia de Produção

  
Prof. Associado **GERALDO ROBERTO MARTINS DA COSTA**  
Presidente da Comissão da Pós-Graduação da EESC

## **DEDICATÓRIA**

A Ingrid, por todo amor, compreensão, apoio e paciência ao longo da elaboração deste trabalho.

## **AGRADECIMENTOS**

Aos meus pais por todo o esforço pela minha formação;

À minha avó Nelly pela ajuda sempre quando necessário;

Ao Professor Marcelo Seido Nagano, pela orientação, conselhos e amizade;

Aos Professores Alceu Camargo e Daisy Rebelatto pelas sugestões, críticas e conselhos;

Ao amigo Marcelo Botelho pela amizade e incentivo a começar o mestrado em engenharia de produção.

Ao Panqueca pelo companheirismo nas madrugadas de trabalho.

## RESUMO

MARQUES, F. T. **Otimização de carteiras com lotes de compra e custos de transação, uma abordagem por algoritmos genéticos.** 2007. 88f. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2007.

Um dos problemas fundamentais em finanças é a escolha de ativos para investimento. O primeiro método para solucionar este problema foi desenvolvido por Markowitz em 1952 com a análise de como a variância dos retornos de um ativo impacta no risco do portfólio no qual o mesmo está inserido. Apesar da importância de sua contribuição, o método desenvolvido para a otimização de carteiras não leva em consideração características como a existência de lotes de compra para os ativos e a existência de custos de transação. Este trabalho apresenta uma abordagem alternativa para o problema de otimização de carteiras utilizando Algoritmos Genéticos. Para tanto são utilizados três algoritmos, o Algoritmo Genético Simples, o Algoritmo Genético Multiobjetivo (*Multi Objective Genetic Algorithm* - MOGA) e o Algoritmo Genético de Ordenação Não Dominante (*Non Dominated Sorting Genetic Algorithm* - NSGA II).

O desempenho apresentado pelos Algoritmos Genéticos neste trabalho mostram a perspectiva para a solução desse problema tão importante e complexo, obtendo-se soluções de alta qualidade e com menor esforço computacional.

Palavras-Chave: Otimização de Carteiras, Markowitz, Algoritmos Genéticos



## **ABSTRACT**

MARQUES, F. T. **Portfolio Optimization with round lots and transaction costs, an approach with genetic algorithms.** 2007. 88p. Dissertation (Master) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2007.

One of the basic problems in finance is the choice of assets for investment. The first method to solve this problem was developed by Markowitz in 1952 with the analysis of how the variance of the returns of an asset impacts in the portfolio risk in which the same is inserted. Despite the importance of its contribution, the method developed for the portfolio optimization does not consider characteristics as the existence of round lots and transaction costs. This work presents an alternative approach for the portfolio optimization problem using Genetic Algorithms. For that three algorithms are used, the Simple Genetic Algorithm, the Multi Objective Genetic Algorithm (MOGA) and the Non Dominated Sorting Genetic Algorithm (NSGA II).

The performance presented for the Genetic Algorithms in this work shows the perspective for the solution of this so important and complex problem, getting solutions of high quality and with lesser computational effort.

Keywords: Portfolio Optimization, Markowitz, Genetic Algorithms.

## Lista de Figuras

FIGURA 1: POSSÍVEIS SOLUÇÕES	8
FIGURA 2: FRONTEIRA EFICIENTE	12
FIGURA 3: DIVISÃO DOS RISCOS	12
FIGURA 4: VAR E CVAR	19
FIGURA 5: DISTRIBUIÇÃO DOS RETORNOS	23
FIGURA 6: FUNCIONAMENTO DE UM ALGORITMO GENÉTICO	32
FIGURA 7 NSGA II	39
FIGURA 8: DISTRIBUIÇÃO DOS RESULTADOS POR ALGORITMO	52
FIGURA 9: DISTRIBUIÇÃO DOS RESULTADOS POR TAXA DE MUTAÇÃO	53
FIGURA 10: DISTRIBUIÇÃO DOS RESULTADOS POR TAMANHO DE POPULAÇÃO	54
FIGURA 11: DISTRIBUIÇÃO DOS RESULTADOS POR ALGORITMO E TAMANHO DE POPULAÇÃO	56

## Lista de Tabelas

Tabela 1: VaR e CvaR	20
Tabela 2: Ações do IBrX-50	41
Tabela 3: Ações para cada problema	44
Tabela 4: Custos de Transação	46
Tabela 5: Mudança de investimentos entre períodos	46
Tabela 6: Lotes de Transação	47
Tabela 7: Resultados	51
Tabela 8: Tabela ANOVA	55

## **Lista de Siglas**

**AG** Algoritmo Genético

**ANOVA** Análise de Variância

**ARCH** Modelo Heterocedasticidade Auto Regressiva

**ARMA** modelo auto regressivo e de médias móveis

**CVaR** Conditional Value at Risk

**GARCH** Modelo Heterocedasticidade Auto Regressiva Generalizado

**IGARCH** Modelo Heterocedasticidade Auto Regressiva Generalizado Integrado

**MAD** Desvio Absoluto Médio

**MOGA** Multi Objective Genetic Algorithm

**nc** Contador de Nichos

**NSGA II** Non Dominated Sorting Genetic Algorithm

**PSO** Particle Swarm Optimization

**VaR** Value at Risk

<b>LISTA DE FIGURAS</b>	<b>III</b>
<b>LISTA DE TABELAS</b>	<b>IV</b>
<b>LISTA DE SIGLAS</b>	<b>V</b>
<b>1-INTRODUÇÃO</b>	<b>2</b>
1.1. LOCALIZAÇÃO	2
1.2. PROBLEMA	3
<b>2- OTIMIZAÇÃO DE CARTEIRAS</b>	<b>7</b>
2.1-OTIMIZAÇÃO MULTIOBJETIVO	7
2.2-MODELO DE MARKOWITZ	10
2.3-ESTUDOS POSTERIORES	13
A) MODELO DESVIO ABSOLUTO MÉDIO (MAD)	14
B) MODELO <i>VALUE AT RISK</i>	16
C) MODELO <i>CONDITIONAL VALUE AT RISK</i>	18
<b>3-ALGORITMOS GENÉTICOS</b>	<b>26</b>
3.1-PARÂMETROS DO ALGORITMO	32
3.2-ALGORITMOS GENÉTICOS MULTIOBJETIVOS	34
3.2.1- MOGA ( <i>MULTI OBJECTIVE GENETIC ALGORITHM</i> )	35
3.2.2-NSGA II ( <i>NON DOMINATED SORTING GENETIC ALGORITHM</i> )	38
<b>4-MÉTODO DE PESQUISA</b>	<b>40</b>
4.1-REPRESENTAÇÃO DO PROBLEMA	41
4.2-CRUZAMENTO E MUTAÇÃO	42
4.3-QUANTIDADE DE ATIVOS DISPONÍVEIS	43
4.4-POPULAÇÃO INICIAL	44
4.5-CUSTOS DE TRANSAÇÃO	45
4.3-LOTES DE COMPRA	47
4.4- PREVISÃO DOS RETORNOS, VOLATILIDADES E COVARIÂNCIAS	48
<b>5-ANÁLISE DOS RESULTADOS</b>	<b>50</b>
<b>6-CONSIDERAÇÕES FINAIS</b>	<b>57</b>
<b>7-REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>58</b>
<b>ANEXO 1 – DESENVOLVIMENTO DO MODELO DE MARKOWITZ</b>	<b>62</b>

<b><u>ANEXO 2 - ALGORITMO DE PREPARAÇÃO DOS DADOS</u></b>	<b>65</b>
<b><u>ANEXO 3 - ALGORITMO GENÉTICO SIMPLES</u></b>	<b>67</b>
<b><u>ANEXO 4 - MOGA</u></b>	<b>71</b>
<b><u>ANEXO 5 NSGA</u></b>	<b>76</b>

# 1-INTRODUÇÃO

## 1.1. *Localização*

O mercado financeiro passou por uma grande revolução nas últimas décadas com a disseminação de métodos quantitativos avançados para lidar com vários aspectos na tomada de decisão. Métodos estes que não são apenas o diferencial, mas sim requisitos básicos para a sobrevivência em um ambiente extremamente competitivo.

O advento destes métodos é fruto de dois fatores principais: a necessidade de controlar os riscos de diferentes naturezas, provenientes do desenvolvimento de produtos financeiros cada vez mais sofisticados como os derivativos, e a disponibilidade de recursos computacionais com custo relativamente reduzido, tornando possível a aplicação de diversas técnicas em problemas de finanças, com a obtenção de respostas na velocidade que o mercado exige.

Um dos problemas fundamentais em finanças é a escolha de ativos para investimento. Todos os agentes do mercado, sejam indivíduos ou empresas, enfrentam constantemente a questão de onde investir seus recursos dentre a enorme quantidade de alternativas disponíveis. Para decidir a melhor combinação dessas opções é necessário levar em conta tanto o retorno que será obtido como o risco que será incorrido.

O trabalho feito por Markowitz em 1952 é um marco nas finanças pois desenvolveu um método para solucionar de maneira direta a escolha dos ativos para investimento, suas idéias fazem parte do início do que é chamada Moderna Teoria de Finanças. A contribuição fundamental de Markowitz foi a definição de como a variância dos retornos de um ativo impacta no risco do portfólio no qual o mesmo está inserido. Sob o ponto de vista matemático a abordagem de Markowitz para a otimização de

carteiras é um problema de Pesquisa Operacional, especificamente um problema de Otimização Quadrática.

Apesar da importância de sua contribuição, o método desenvolvido para a otimização de carteiras não leva em consideração características do mundo real, como por exemplo a existência de lotes de compra para os ativos (em seu modelo os ativos são infinitamente divisíveis, ou seja, é possível comprar ou vender qualquer quantidade) e a existência de custos de transação (no momento da compra ou venda de ativos ocorre a incidência de custos de corretagem). Este trabalho apresenta uma abordagem alternativa para o problema utilizando Algoritmos Genéticos.

## **1.2. Problema**

A necessidade de otimizar os recursos, dada a grande variedade de opções de investimento disponíveis, fez com que a otimização de carteiras se tornasse um tópico amplamente estudado.

Dada a miríade de opções de investimento e o acirramento da competição em praticamente todos os setores, a alocação de recursos tem que ser feita de maneira mais eficiente possível, balanceando o *trade-off* entre maximizar o retorno e minimizar o risco.

A importância deste trabalho consiste na otimização de carteiras com a inserção de duas restrições do mundo real, lotes de transação mínimos para cada ativo e custos para a compra e venda destes ativos. Estas restrições não são consideradas conjuntamente nos modelos encontrados na literatura. Isso se deve ao fato que tais restrições adicionam uma grande complexidade para o problema, tornando-o praticamente intratável analiticamente. Para solucionar este problema uma ferramenta



de sistemas inteligentes será aplicada de modo a contornar essa complexidade, os Algoritmos Genéticos.

A aplicação dos Algoritmos Genéticos para esse tipo de problema tem como grande vantagem o fato que o desenvolvimento formal do modelo, com todo seu tratamento matemático, é desnecessário. O algoritmo genético necessita apenas de uma função que avalie a qualidade das soluções (também chamada de *fitness*), que no caso de otimização de carteiras são duas as variáveis que determinam a qualidade da solução, o risco e o retorno da carteira.

Para lidar com esses dois objetivos conflitantes também existe uma categoria de Algoritmos Genéticos que será considerada no trabalho: os Algoritmos Genéticos multiobjetivos. Neste trabalho serão utilizados três algoritmos, o Algoritmo Genético Simples, o Algoritmo Genético Multiobjetivo (*Multi Objective Genetic Algorithm - MOGA*) e o Algoritmo Genético de Ordenação Não Dominante (*Non Dominated Sorting Genetic Algorithm - NSGA II*).

### **1.3. Objetivo da Pesquisa**

O objetivo principal da pesquisa é analisar o desempenho de Algoritmos Genéticos na otimização de carteiras, tendo como objetivos secundários:

1. Verificar a estabilidade dos resultados obtidos pelos algoritmos e a possível necessidade de ajustes específicos para lidar com o problema;
2. Comparar o desempenho dos Algoritmos Genéticos Multiobjetivos e do algoritmo genético simples e verificar os seus desempenhos para o problema de otimização de carteiras.

## **1.4. Justificativa**

O modelo desenvolvido por Markowitz (1952) para o problema de otimização de carteiras pode ser considerado um dos pilares fundamentais da Moderna Teoria de Finanças e, segundo Nabholz (2006), o trabalho de Markowitz é baseado em oito hipóteses:

*Hipótese 1:* Os investidores avaliam carteiras apenas com base no valor esperado e nas variâncias das taxas de retorno dos ativos disponíveis;

*Hipótese 2:* Os investidores buscam maximizar seu retorno. Quando postos a escolher entre duas carteiras com mesmo risco, sempre escolherão a com maior retorno;

*Hipótese 3:* Os investidores buscam minimizar seu risco. Quando postos a escolher entre duas carteiras com mesmo retorno, sempre escolherão a com menor risco;

*Hipótese 4:* Os ativos são infinitamente divisíveis, sendo possível comprar qualquer fração do ativo;

*Hipótese 5:* Existe uma taxa livre de risco, na qual o investidor pode tanto aplicar, como tomar recursos emprestados;

*Hipótese 6:* O volume das operações do investidor não afeta os preços de mercado;

*Hipótese 7:* Os custos operacionais de comprar ativos são irrelevantes.

Algumas destas hipóteses foram utilizadas para “simplificar” a realidade e permitir o desenvolvimento analítico do modelo.

Este trabalho busca contribuir para o modelo de Markowitz de modo que as hipóteses 4 e 7 não sejam necessárias, dado que estas duas hipóteses não correspondem

à realidade de um investidor no mercado financeiro, tornando o modelo mais generalista.

### **1.5. Delimitações do estudo**

Este trabalho é baseado em modelos computacionais, onde a necessidade de cálculos cresce exponencialmente com o aumento de variáveis. Portanto algumas limitações precisam ser feitas quanto ao número de ativos considerados e os níveis nas variáveis das configurações dos algoritmos:

- a) Serão considerados 50 ativos disponíveis para investimento;
- b) A quantidade de níveis nos fatores que influenciam o desempenho do algoritmo foi estabelecida de modo que todas as combinações possíveis entre os níveis dos fatores fossem realizadas.

### **1.6. Estrutura do trabalho**

O trabalho é dividido em cinco seções. Na seção 2 são descritos os principais conceitos que fundamentam a otimização de carteiras, o modelo desenvolvido por Markowitz e os modelos desenvolvidos recentemente. Na seção 3 são apresentados os Algoritmos Genéticos e os conceitos que fundamentam esse método de otimização, assim como os Algoritmos Genéticos Multiobjetivos. Na seção 4 é apresentado o método de pesquisa do processo de experimentação computacional e seus resultados obtidos. Na seção 5 são apresentadas as análises dos resultados obtidos e, finalizando na seção 6, as considerações finais.

## **2- OTIMIZAÇÃO DE CARTEIRAS**

Neste capítulo são apresentados os principais pilares teóricos nos quais se fundamenta este trabalho.

Primeiramente são apresentados os princípios da otimização multiobjetivo, em seguida é apresentado o modelo de Markowitz para otimização de carteiras e os modelos alternativos para otimização de carteiras .

### ***2.1-Otimização Multiobjetivo***

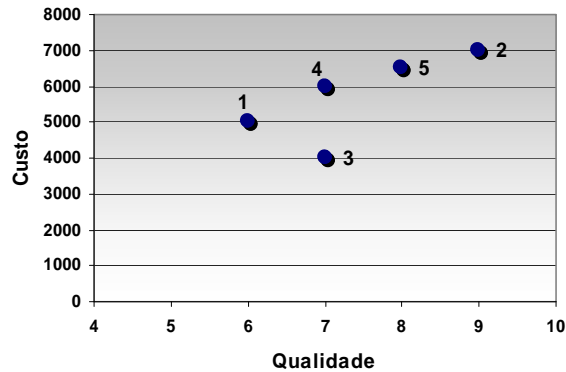
Um problema de otimização é composto de uma função objetivo e um conjunto de restrições, ambos relacionados às variáveis de decisão. O problema pode ser de minimização ou de maximização da função objetivo.

A resposta para o problema, ou seja, o ótimo global, será o menor (ou maior) valor possível para a função objetivo para o qual o valor atribuído às variáveis não viole nenhuma restrição.

Um problema de otimização multiobjetivo também é composto de restrições, mas apresenta um conjunto de funções objetivo que necessitam ser trabalhadas conjuntamente.

Um exemplo de problema com mais de um objetivo seria o projeto de um prédio onde se deseja minimizar o custo da obra e maximizar a percepção de qualidade dos futuros compradores. À medida que se reduz o custo também diminui a percepção

de qualidade. Portanto, não existe uma única solução ótima e sim um conjunto de soluções.



**Figura 1:** Possíveis Soluções

O objetivo neste caso é minimizar o custo e maximizar a qualidade. Considerando as cinco opções para o problema, conforme mostrado na figura 1, a opção 4 é descartada pois oferece uma qualidade nível 7 por um custo de \$60.000, enquanto a opção 3 oferece o mesmo nível de qualidade por \$40.000. A opção 1 é descartada pelo mesmo motivo. Sobram então as opções 3, 5 e 2 como boas alternativas.

Uma solução domina uma outra se seus valores são melhores em todos os objetivos ou pelo menos o valor em um objetivo é melhor e os outros são iguais. No caso acima a opção 3 domina tanto a opção 1 como a opção 4. As opções 5 e 2 não são dominadas por nenhuma outra, assim como a 3. A linha contendo todas as soluções não dominadas é conhecida como *Fronteira de Pareto*.

Para este caso, uma solução  $x$  domina uma solução  $y$  ( $x \succcurlyeq y$ ) se:

- 1- A solução  $x$  não é pior que  $y$  em nenhum objetivo;
- 2- Pelo menos um objetivo de  $x$  é melhor que  $y$ .

No caso  $3 \succcurlyeq 1$  e  $3 \succcurlyeq 4$ . O conceito de dominância permite avaliar soluções com múltiplos objetivos.

Com relação a este caso, algumas propriedades podem ser estabelecidas (ARAÚJO, 1984):

- 1- Não reflexiva. Uma solução não pode dominar a si mesma;
- 2- Não simétrica.  $x \succcurlyeq y$  não implica em  $y \succcurlyeq x$ ;
- 3- Transitividade. Se  $x \succcurlyeq y$  e  $y \succcurlyeq z$  então  $x \succcurlyeq z$ .

O conjunto de soluções não dominadas para todo o espaço de busca factível é chamado conjunto ótimo de Pareto.

Quando a informação adicional sobre a importância dos objetivos é desconhecida, todas as soluções Pareto-ótimas são igualmente importantes. Deb (2001) assinala duas importantes metas em Otimização Multiobjetivo:

- 1- Encontrar um conjunto de soluções o mais próximo possível da *Frenteira de Pareto*.

- 2- Encontrar um conjunto de soluções com a maior diversidade possível. A primeira meta é comum para qualquer processo de otimização.

Soluções muito distantes da *Frenteira de Pareto* não são desejáveis. Porém, encontrar a maior diversidade dentro das soluções é uma meta específica para otimização multiobjetivo.

Em relação ao problema de pesquisa, a otimização de carteiras consiste na escolha de ativos de modo a obter o maior retorno dado um nível de risco, ou o menor risco dado um nível de retorno exigido.

A seleção correta de oportunidades de investimento, entre os inúmeros ativos disponíveis no mercado financeiro, é tarefa constante no dia a dia das organizações. A distribuição das aplicações entre os ativos é o que é chamada carteira, ou portfólio.

A otimização de carteiras, assim como muitos dos problemas do mundo real que envolvem múltiplas medidas de avaliação, possui dois objetivos conflitantes a serem considerados no processo de otimização: o retorno e o risco.

A *Fronteira de Pareto*, em otimização de carteiras, é o conjunto de opções de investimento que ofereçam o maior retorno para diferentes opções de risco.

Em 1952 Harry Markowitz, em seu trabalho pioneiro, propôs um modelo formal para determinar uma carteira ótima. O modelo de Markowitz (1952) quantifica o problema utilizando apenas duas variáveis: os retornos e a matriz de covariâncias como a medida do risco. Sendo o risco definido como o grau de incerteza associado a um evento.

## **2.2-Modelo de Markowitz**

O modelo desenvolvido por Markowitz (1952) estabelece uma estratégia de investimento baseada em maximizar o retorno dado um nível de risco previamente especificado, ou minimizar o risco dado um nível de retorno. Também conhecido como modelo “média-variância” parte do princípio que, para o investidor, o retorno esperado e a volatilidade dos prováveis retornos são aspectos cruciais na definição do portfólio ótimo. Para este modelo são utilizadas as medidas estatísticas de valor esperado e variância da distribuição dos retornos para descrever, respectivamente, o retorno e o risco do investimento.

A determinação da decisão indicada pelo modelo é obtida utilizando a média e a covariância dos retornos entre os diversos ativos disponíveis no mercado. A formulação matemática do modelo pode ser descrita como:

-Minimizar a variância do Portifólio, dada por:

$$\sigma = \sum_{i=1}^N \sum_{j=1}^N w_i w_j \sigma_{ij}$$

ou,

-Maximizar o retorno do Portifólio, dado por:

$$R = \sum_{i=1}^N w_i r_i$$

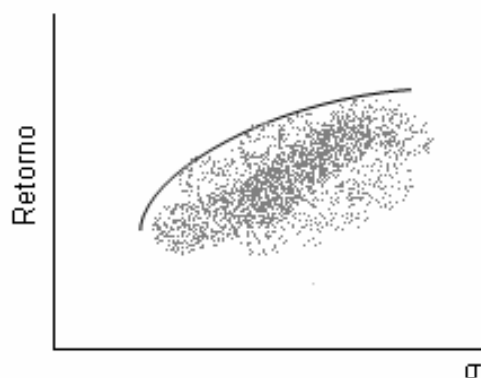
ambos sujeitos a:  $\sum_{i=1}^N w_i = 1$  e  $0 \leq w_i \leq 1$ ,  $i = 1, \dots, N$

Sendo R o retorno da carteira,  $r_i$  retorno do ativo  $i$ ,  $w_i$  a proporção investida no ativo  $i$  e  $\sigma_{ij}$  a covariância entre os ativos  $i$  e  $j$ .

Como fica explícito acima, o problema de otimização de um portfólio tem dois objetivos competindo entre si, maximizar o retorno e minimizar a variância do portfólio. Para esse problema o comum é, ou definir o nível de risco (variância) do portfólio e encontrar a composição que forneça o retorno máximo, ou definir o retorno e determinar a composição que corresponda ao risco mínimo, ou seja, para o investidor o retorno é algo desejável, já a variância não.

Obtendo, associado a cada um dos níveis de retorno, a composição da carteira de menor risco (ou para cada um dos níveis de risco, a composição da carteira com maior retorno), pode-se então traçar uma curva com a relação risco *versus* retorno, denominada fronteira eficiente, conforme pode ser observado na Figura 2.



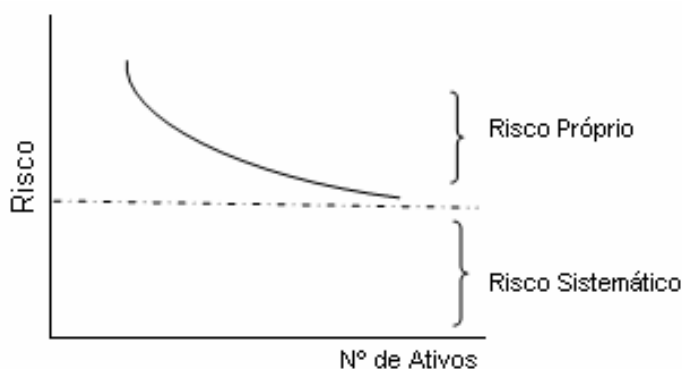


**Figura 2:** Fronteira Eficiente

A linha traçada na Figura 2 é a fronteira eficiente (a *Fronteira de Pareto*), onde estão as melhores soluções.

O modelo desenvolvido por Markowitz também traz a idéia da diversificação, ou seja: "não coloque todos os ovos na mesma cesta", partindo do princípio que a variância da soma dos ativos é menor, ou no máximo igual, à soma das variâncias dos ativos.

Pela Figura 3 pode-se observar que conforme a quantidade de ativos for aumentando o risco da carteira se estabiliza em torno de um valor, o risco sistemático. Este risco é o qual estão expostos todos os ativos, não sendo possível mitigar este risco através da aplicação em outros ativos. Já o inverso ocorre com o risco próprio, que é característico do ativo em que se está investindo. Este risco pode ser mitigado através do investimento em um ativo que tenha correlação negativa com o ativo.



**Figura 3:** Divisão dos Riscos

A determinação da fronteira eficiente é apenas uma parte do processo de escolha de investimentos, sendo a aversão ao risco e as curvas de indiferença do investidor os fatores que irão determinar, em conjunto com a fronteira eficiente obtida, quais os ativos serão escolhidos para investimento. O foco deste trabalho é apenas no modelo de Markowitz para a determinação da fronteira eficiente, as curvas de indiferença e a aversão ao risco dos investidores não serão tratadas neste trabalho. Para maior entendimento do modelo de Markowitz, o mesmo é apresentado no Apêndice 1.

### ***2.3-Estudos Posteriores***

Muitas pesquisas foram feitas na área de otimização de portfólios mas, em todo o levantamento bibliográfico, não foi encontrado modelo de otimização de carteira que invalidasse os conceitos postulados por Markowitz (1952). Devido à importância deste trabalho foi concedido ao autor o prêmio Nobel em Economia no ano de 1990.

A proposição de Markowitz permitiu que os agentes do mercado, pela primeira vez, utilizassem os conceitos de risco e retorno de forma conjunta na avaliação de investimentos. Apesar da grande aceitação e disseminação o modelo de Markowitz tem sofrido algumas críticas. Dentre as críticas ao modelo, está a utilização da variância como medida de risco. Apesar de ser a medida de risco mais popular, a variância não é a única medida de risco, podendo ser substituída por outras medidas de risco.

Variabilidade dos retornos, quando positivos, não devem ser penalizados, pois investidores se preocupam com baixos rendimentos do portfólio, e não com os altos. A variância, que mede a dispersão dos dados ao redor de sua média, penaliza o investidor tanto pelos possíveis ganhos como pelas possíveis perdas (BRADLEY e TAQQU, 2004).

### a) Modelo Desvio Absoluto Médio (MAD)

Konno (1991) propôs um modelo de otimização de carteiras que utiliza como medida de risco o desvio absoluto médio. Segundo Ribeiro *et al* (2003) o modelo de Konno é apontado na literatura de finanças como uma importante contribuição para a resolução de problemas de gestão de carteiras por introduzir uma medida de risco mais simples do que a utilizada por Markowitz.

O modelo considera que as incertezas com relação aos retornos dos ativos são representadas de forma discreta por meio de cenários, de forma que o retorno do ativo  $i$  no cenário  $S$  seria representado por  $R_{is}$ , e que o retorno do ativo  $i$  seria dado por :

$$\mu_i = \frac{\sum_{s=1}^S R_{is}}{S}$$

O desvio absoluto médio da carteira  $x=(x_1, \dots, x_n)$  seria dado por :

$$W(x) = \frac{1}{S} \sum_{s=1}^S \left| \sum_{i=1}^n (R_{is} - \mu_i) x_i \right|$$

Matematicamente, a formulação do problema de otimização de carteiras proposto por Konno seria:

$$\text{Minimizar } Z = \frac{1}{S} \sum_{s=1}^S Y_s$$

sujeito a:

$$\begin{aligned}
Y_s &\geq -\sum_{i=1}^N (R_{is} - \mu_i)x_i \\
Y_s &\geq \sum_{i=1}^N (R_{is} - \mu_i)x_i \\
\sum_{i=1}^N x_i \mu_i &= \rho \\
\sum_{i=1}^N x_i &= 1 \\
x_i &\geq 0
\end{aligned}$$

onde:

S - número de cenários utilizados para representar as incertezas com relação aos retornos dos ativos candidatos a compor o portfólio;

$Y_s$  - variável auxiliar utilizada na modelagem do desvio absoluto médio;

N - número de ativos candidatos a compor o portfólio;

$R_{is}$  - retorno do i-ésimo ativo candidato a compor o portfólio no cenário s;

$\mu_i$  - valor esperado dos retornos do i-ésimo ativo candidato a compor o portfólio;

$x_i$  - fração do capital a ser aplicado no ativo candidato i;

$\rho$  - valor esperado dos retornos do portfólio (valor requerido pelo investidor).

A função objetivo em conjunto com os dois primeiros conjuntos de restrições, modelam o desvio absoluto médio dos retornos do portfólio, que deve ser minimizado.

A terceira restrição representa o valor esperado do retorno do portfólio. A variável  $\rho$  é o valor desejado pelo investidor (dado de entrada para o modelo). A penúltima restrição garante que todo o capital disponível seja investido, e a última restrição assegura a não existência de investimento negativo. Konno destaca como vantagem da formulação MAD, quando comparada com o modelo média-variância de Markowitz, os seguintes pontos:

- o modelo MAD não requer a estimação da matriz de covariâncias;

- o modelo MAD é linear, o que faz com que sua solução seja mais rápida e eficiente do que a solução do modelo quadrático de Markowitz;
- o modelo MAD automaticamente limita o número de ativos no portfólio em  $2S + 2$  (número de restrições do problema), mesmo se o número de ativos candidatos for muito maior. Tal fato pode implicar em um menor custo de transação quando da revisão do portfólio.

Segundo Ribeiro *et al* (2003) o modelo de Markowitz possui, como maior desvantagem, o fato de recair num problema de otimização quadrática. Por sua vez o modelo de Konno minimiza o valor absoluto o que, devido à estrutura do problema, pode ser transformado num problema de programação linear.

Além disso, Konno (1991) demonstra que se os retornos dos ativos seguirem uma distribuição normal multivariada, a minimização do desvio absoluto médio é equivalente à minimização da variância.

Apesar de todas estas vantagens Ribeiro *et al* (2003) realizaram um estudo comparativo com ações do mercado português e encontraram um desempenho superior no modelo de Markowitz, apesar do maior custo computacional.

## **b) Modelo *Value at Risk***

A substituição da variância por alguma medida de risco baseada em quantis (como o *Value at Risk* ou *Conditional Value at Risk*) tem tido espaço entre diversos autores. O *Value at Risk* (VaR) tem como informação a pior perda possível com um nível de significância e horizonte definidos, por exemplo a pior perda em uma semana com 95% de certeza.

O VaR foi inicialmente aplicado a investimentos, como ações, no começo da década de 90 (JORION, 1999) e posteriormente passou a ser utilizado como métrica para diversos tipos de riscos, como risco de crédito e operacional, entre outros, tornando-se portanto, o padrão para uma enorme gama de empresas.

Utilizando a estimação pelo método Delta-Normal, método este que assume que os retornos tenham distribuição normal padronizada, o VaR de uma carteira será dado por (Jorion, 1999):

$$VaR_p = \alpha \sigma_p W = \alpha \sqrt{x' \Sigma x},$$

onde  $\alpha$  é o nível de significância desejado e  $\Sigma$  é a matriz de covariância dos retornos.

Matematicamente, o problema de otimização de portfólio cuja função objetivo seja a minimização do VaR a um dado nível de confiança  $\beta$  %, sujeito ao atendimento a um dado valor esperado mínimo, pode ser formulado da seguinte forma:

Minimizar  $\alpha$

Sujeito a

$$-\sum_{i=1}^N x_i R_{is} - MY_s \leq \alpha$$

$$\sum_{s=1}^S Y_s = [(1 - \beta\%)S]$$

$$\sum_{i=1}^N x_i \mu_i = \rho$$

$$\sum_{i=1}^N x_i = 1$$

$$x_i \geq 0$$

$$Y_s \in \{0,1\}$$

onde:

$\alpha$ - variável que representa o VaR ao nível de confiança  $\beta\%$ ;

N - número de ativos candidatos a compor o portfólio;

$x_i$  - fração do capital a ser aplicado no ativo candidato  $i$ ;

$r_{is}$  - retorno do  $i$ -ésimo ativo candidato a compor o portfólio no cenário  $s$ ;

M- número muito grande ( $M \rightarrow +\infty$ );

$y_s$  - variável auxiliar para o cálculo do VaR;

S - número de cenários utilizados para representar as incertezas com relação aos; retornos dos ativos candidatos a compor o portfólio;

$\beta\%$ - nível de confiança para o cálculo do VaR

$\mu_i$  - valor esperado dos retornos do  $i$ -ésimo ativo candidato a compor o portfólio

$\rho$  - valor esperado dos retornos do portfólio (valor requerido pelo investidor)

A função objetivo em conjunto com o primeiro e segundo conjunto de restrições, modelam o VaR ao nível de confiança  $\beta \%$ , que deve ser minimizado. O VaR ao nível de confiança  $\beta \%$  está associado a um nível de perda que só é superado por  $(1 - \beta)\%$  dos cenários.

Apesar de o VaR ser uma medida amplamente aceita, seja pelos reguladores como pelos participantes do mercado financeiro, ele apresenta algumas limitações. A principal limitação é sobre sua qualidade como medida de risco.

### **c) Modelo *Conditional Value at Risk***

Artzner et alli (1997) propuseram quatro propriedades necessárias para que uma medida de risco,  $\pi$ , possa ser considerada coerente, dada uma variável aleatória X:

1-Monotonicidade:  $X \leq Y, \pi(X) \leq \pi(Y)$  ;

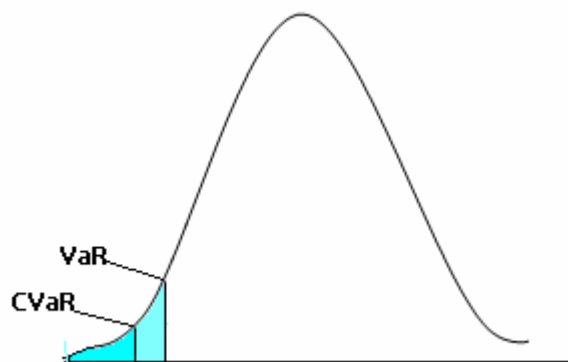
2-Translação:  $\pi(X+a) = \pi(X) + a$  ;

3-Homogeneidade:  $\pi(\lambda X) = \lambda\pi(X)$ ;

4-Sub-aditividade:  $\pi(X+Y) \geq \pi(X) + \pi(Y)$ .

O VaR não apresenta a quarta propriedade, sub-aditividade, isto é, o VaR da soma de dois ativos pode ser maior que a soma dos VaRs de cada ativo. Uma medida proposta pelos autores é o *Conditional Value at Risk* (CVaR), que é definido como a perda esperada dado que essa perda foi maior que o VaR, ou seja:  $E(X|X < VaR)$ .

A Figura 4 a seguir exemplifica o conceito do CVaR. Nela está uma distribuição de retornos e seu respectivo VaR. Como o CVaR é a perda esperada, ou perda média, dado que esta perda foi menor que o VaR, perdas estas representadas pela área preenchida.



**Figura 4:** VaR e CVaR



Se a distribuição dos dados for normal então o valor dos percentis será (Jorion, 1999):

Percentil	90%	95%	99%
$\alpha$	1,28	1,65	2,32
$E(X/X < \alpha)$	1,75	2,06	2,67

**Tabela 1:** VaR e CvaR

O problema de otimização de portfólio cuja função objetivo seja a minimização do CVaR a um dado nível de confiança  $\beta$  %, sujeito ao atendimento a um dado valor esperado mínimo, pode ser escrito da seguinte :

$$\text{Minimizar } Z = \alpha + \frac{1}{(1-\beta)S} \sum_{s=1}^S u_s$$

Sujeito a

$$\begin{aligned} u_s &\geq 0 \\ u_s &\geq -\sum_{i=1}^N x_i R_{iS} - \alpha \\ \sum_{i=1}^N x_i \mu_i &= \rho \\ \sum_{i=1}^N x_i &= 1 \\ x_i &\geq 0 \end{aligned}$$

onde:

$\alpha$ - variável que fornece o VaR do portfólio a nível de confiança  $\beta$ %;

$\beta$  - nível de confiança para o cálculo do VaR e do CVaR;

S - número de cenários utilizados na representação das incertezas com relação aos retornos dos ativos candidatos a compor o portfólio;

$u_s$  - variável auxiliar para o cálculo do CVaR;

$N$  - número de ativos candidatos a compor o portfólio;

$x_i$  - fração do capital a ser aplicado no ativo candidato  $i$ ;

$r_{is}$  - retorno do  $i$ -ésimo ativo candidato a compor o portfólio no cenário  $s$ ;

$\mu_i$  - valor esperado dos retornos do  $i$ -ésimo ativo candidato a compor o portfólio;

$\rho$  - valor esperado dos retornos do portfólio (valor requerido pelo investidor).

A função objetivo e os dois primeiros conjuntos de restrições modelam o CVaR do portfólio a nível de confiança  $\beta$  %. A terceira restrição garante a obtenção do valor esperado requerido pelo investidor. A quarta restrição garante o investimento total. A quinta restrição garante que não haja investimento negativo.

Bradley e Taqqu (2004) utilizam a teoria de Valores Extremos para estimar o Value at Risk da carteira e depois utilizá-lo como medida de risco na otimização, já Luthi e Doege (2005) utilizam o Conditional Value at Risk. Dentcheva e Ruszczyński (2006) estabelecem a relação entre os conceitos de VaR e CVaR e dominância estocástica e os utilizam na otimização de carteiras de um agente avesso ao risco.

Pirvu (2005) busca a otimização de portfólio com restrição do Value at Risk, ou seja, o risco máximo é estipulado e o foco passa a ser o máximo retorno possível. Kalin e Zagst (1999) derivam as medidas de VaR e variância para outros tipos de distribuição, como a Distribuição Hiperbólica Generalizada, e mostram sua aplicabilidade no problema de otimização de carteiras.

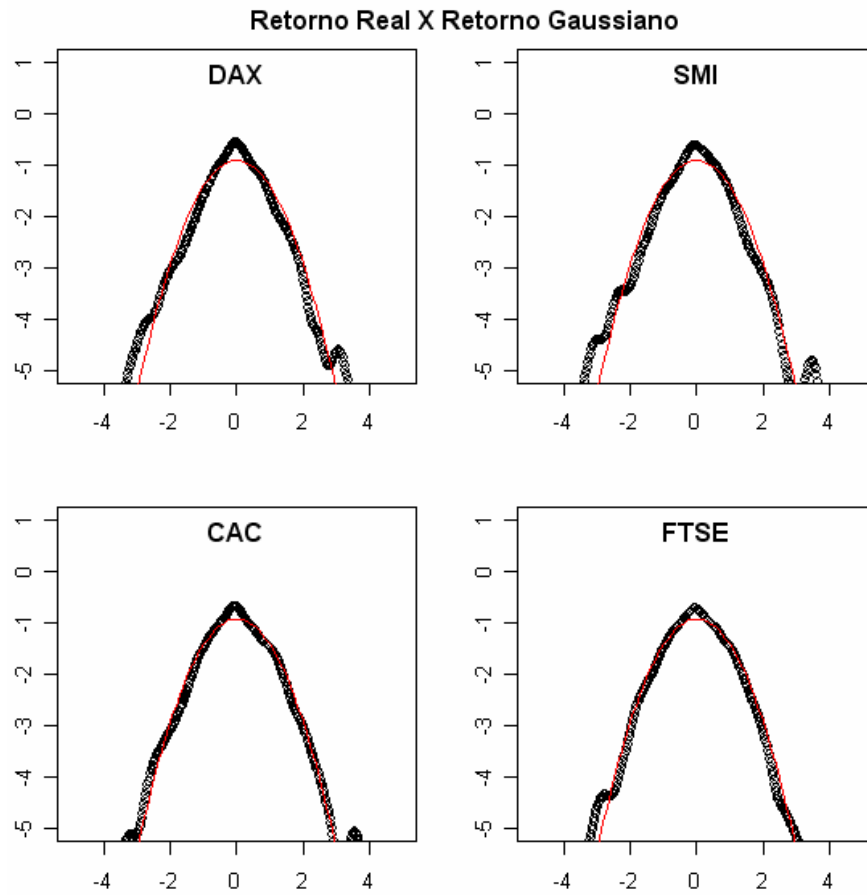
Apesar da importância destas medidas de risco, Di Giorgi (2002) faz um estudo comparativo de otimização de portfólios considerando três medidas de risco (Variância, VaR e CVaR) e mostra que a fronteira eficiente obtida quando o VaR e o CVaR são

utilizados é apenas um subgrupo da fronteira que seria obtida se fosse utilizada a Variância como medida de risco.

Além da medida de risco utilizada, em muitos estudos é avaliada a performance da otimização quando são consideradas diversos tipos de distribuições para os retornos (a variável aleatória da otimização de carteiras).

Dentro desta categoria, não considerando adequada a suposição da distribuição normal multivariada para os retornos, Di Clemente e Romano (2003) utilizam as distribuições não elípticas para os retornos, Kato (2004) analisa a questão de otimização de carteiras com a distribuição discreta para os retornos. Já Martin *et al* (2003) utilizam distribuições alfa-estáveis, que englobam tanto a distribuição normal (gaussiana) como distribuições não gaussianas, para capturar o efeito de assimetrias e "caudas pesadas" na distribuição dos fatores de risco. Dentro desta categoria o trabalho de Maringer (2003) considera restrições em medidas de risco da mesma forma que Pirvu (2005) mas vai além e também não considera a hipótese de normalidade sobre a distribuição dos retornos.

O principal problema do pressuposto da distribuição normal dos retornos é com relação ao risco estimado. A distribuição normal (gaussiana) não consegue captar os valores extremos e de baixa probabilidade, conforme por ser observado na figura 5.



**Figura 5:** Distribuição dos Retornos

Pela figura 5 fica claro a não adequação da distribuição gaussiana para representar a distribuição dos retornos de mercados acionários. Nenhum dos quatro índices se ajustou completamente à distribuição gaussiana.

Di Clementi e Romano (2003) comparam os resultados de diferentes tipos de distribuição para ações do mercado italiano e chegam a conclusão que, apesar da distribuição dos retornos não serem normais, a fronteira eficiente obtida com utilização da distribuição normal dos retornos no modelo de Markowitz não apresenta grande diferença das fronteiras obtidas com outras distribuições.

Além da utilização de diferentes distribuições para os retornos dos ativos, diversos autores propuseram métodos para o problema de otimização de carteiras, dentre eles Lauprete *et al* (2002) estudam formas robustas para a estimação dos

parâmetros a serem utilizados na otimização, Gilli e Kellezi (2000) estudam a aplicação do algoritmo "*Threshold Accepting*", Bensalah (2002) e se utilizam da Teoria de Valores Extremos para otimizar o portfólio de um agente com uma aversão ao risco "extrema" .

Dentro dessa categoria, poucos são os estudos utilizando Algoritmos Genéticos. Dentre as aplicações encontradas de Algoritmos Genéticos em Finanças a grande maioria é voltada para aplicações no mercado de capitais, sendo o foco a questão de quando comprar ou não determinado ativo (*Market Timing*) e/ou quais ativos serão utilizados (*Stock Selection*).

Lee *et al* (2005) utilizam os Algoritmos Genéticos e *Particle Swarm Optimization* (PSO) para a questão de *Market Timing*, mostrando que as duas técnicas são adequadas e que, no exemplo que ele utilizou, PSO necessitou de uma quantidade menor de iterações para alcançar o ótimo. Bauer (1994) também utiliza os Algoritmos Genéticos para descobrir "boas regras" para operar no mercado de capitais, se preocupando basicamente com a questão de *Market Timing*.

Já na área de otimização de portfólios existem trabalhos como o de Oh *et al* (2005) onde os Algoritmos Genéticos são utilizados para montar uma carteira de investimentos que tenha o comportamento de determinado índice de mercado, no caso do artigo de Oh *et al* o índice utilizado como referência foi o do mercado coreano.

Xia *et al* (2000) lidam com o problema de otimização de carteiras, agregando o Risco e o Retorno da função objetivo. No trabalho os autores buscam maximizar a diferença entre o retorno e o risco da carteira ponderados por um fator "aversão ao risco".

Para entender melhor como o problema de otimização de carteiras pode ser tratado pelos Algoritmos Genéticos a próxima seção traz os princípios de seu funcionamento.

### 3-ALGORITMOS GENÉTICOS

A otimização tem como objetivo a resolução da alocação de recursos, tipicamente limitados, com o intuito de alcançar determinados objetivos. Considerando que o problema tenha um conjunto discreto de soluções possíveis, a resolução do problema passa para um processo de geração, avaliação e comparação de soluções, num determinado limite de tempo.

Segundo Souza (2007) grande parte desses problemas são classificados na literatura como NP-difíceis, também chamados de “problemas de otimização combinatória”. O uso de métodos exatos nesta categoria de problema é bastante limitado e encontrar soluções ótimas, ou mesmo aproximadas, é um desafio nem sempre fácil de ser obtido.

O desafio é conseguir, em tempo reduzido, soluções tão próximas quanto possível da solução ótima, e como ferramenta para esta tarefa existem as heurísticas.

De acordo com Souza (2007) heurística é definida como sendo uma técnica que procura boas soluções (próximas da otimalidade) a um custo computacional razoável, sem, no entanto, estar capacitada a garantir a otimalidade, bem como garantir quão próximo uma determinada solução está da solução ótima.

Uma heurística pode ser desenvolvida para apenas um tipo de problema ou pode ser mais flexível, utilizada na resolução de uma classe mais ampla de problemas.

Estas heurísticas mais flexíveis, são conhecidas como “metaheurísticas”. Dentre as metaheurísticas destacam-se os Algoritmos Genéticos, Redes Neurais, *Simulated Annealing*, Busca Tabu, Colônia de Formigas etc.

Os Algoritmos Genéticos são modelos computacionais inspirados na evolução encontrada na natureza que trabalham com soluções potenciais para determinado

problema e, através de um método de geração de novas soluções, buscam a solução ótima para o problema.

Segundo Fogel (1995) os organismos vivos podem ser vistos como a dualidade de seu genótipo (o código genético) e seu fenótipo (a resposta do organismo contida no comportamento, fisiologia e morfologia do organismo), ou seja, existem dois espaços, o espaço G para a população de genótipos e o espaço F para a população de fenótipos e quatro funções: epigênese (*fitness*), seleção, cruzamento e mutação.

De acordo com Goldberg (1989) os Algoritmos Genéticos são algoritmos de busca baseados na mecânica da seleção natural. São modelos de processamento computacional que visam simular os mecanismos de seleção natural e evolução encontrados na natureza. A idéia que o algoritmo genético usa como base é a mesma que rege o seguinte princípio que existe na natureza: a sobrevivência dos mais aptos. Sua utilização tem sido cada vez mais explorada principalmente pela robustez e simplicidade que oferece. Os Algoritmos Genéticos foram desenvolvidos na década de 70 por John Holland na Universidade de Michigan.

Os métodos clássicos de otimização iniciam-se com um único candidato, chamado de solução básica, e pelo cálculo de derivadas se determina para qual direção se deve caminhar na busca do próximo candidato. Já os Algoritmos Genéticos transformam uma população de possíveis soluções em uma nova geração de soluções usando os princípios Darwianos de reprodução e sobrevivência dos mais aptos, pela aplicação de operações genéticas tais como cruzamento e mutação.



## **-Representação das Soluções**

Os Algoritmos Genéticos têm como princípio a evolução através de gerações de uma população de indivíduos. Segundo Ávila (2002) os indivíduos nada mais são do que uma possível solução do problema, ou seja, são pontos dispostos dentro do universo de busca da solução ótima.

Um indivíduo ( $X$ ) pode ser representado da seguinte forma, também chamada de *string*:

$$X = [X_1, X_2, \dots, X_n]$$

onde  $X_1$   $X_2$   $X_n$  representam as variáveis que formam o indivíduo, as quais são parâmetros que dependem do problema. O número de variáveis determina a dimensão do espaço de busca.

Segundo Ávila (2002) o número de indivíduos na população é escolhido em função da dificuldade do problema a ser resolvido. Com um número baixo de indivíduos, o universo de busca pode estar sendo representado de maneira muito pobre. Já com um número muito grande de indivíduos, o tempo computacional pode se tornar inviável.

## **-Codificação das Variáveis**

Definido o conceito de indivíduo dentro do Algoritmo Genético é necessário entender como é feita a codificação das variáveis. O ponto de partida para a utilização como ferramenta para solução de problemas, é a representação destes problemas de maneira que os Algoritmos Genéticos possam trabalhar adequadamente sobre eles. Dentre as codificações possíveis, estão:

### ***-Codificação Binária***

Como o próprio nome diz, esta codificação utiliza números binários (apenas conjuntos de 0 e 1) para representar as variáveis. Um indivíduo com codificação binária é representado da seguinte forma:

$$X=[110101101]$$

onde cada variável é representada por um conjunto de bits (genes).

De acordo com Ávila (2002) existem algumas dificuldades em trabalhar com a codificação binária. O principal é a presença de *Hamming cliffs*, que são grandes diferenças nas cadeias de bits que codificam dois números inteiros próximos. Esta dificuldade fica evidente quando, por exemplo, se realiza uma perturbação nos bits mais significativos da variável. Esta perturbação pode causar um grande deslocamento da variável no universo de busca, o que nem sempre é desejado.

### ***-Codificação Real***

Diferentemente da codificação binária a codificação real trabalha diretamente com números reais. Isto é extremamente prático quando se trabalha com variáveis reais por natureza e se usa uma linguagem de programação que lida diretamente com números reais. Como exemplo de indivíduo com codificação real pode-se ter:

$$X=[12,45; 7,88; 9,42]$$

No trabalho de Ávila (2002) é feita uma comparação entre as diferentes metodologias de codificação e conclui-se que não é a codificação das variáveis o responsável maior pelo sucesso dos AGs. O trabalho conclui que é importante que se observe em qual linguagem de programação os AGs serão implementados e, evidente que se a linguagem trabalha diretamente com números binários, a velocidade de processamento dos AGs com código binário será maior. Ao contrário, trabalhando-se com variáveis reais num ambiente de programação tipicamente real, não será necessário efetuar a decodificação das variáveis a cada avaliação da função de mérito.

Os Algoritmos Genéticos são algoritmos de otimização global que empregam uma estratégia de busca paralela e estruturada, mas aleatória. Apesar de aleatórios, o processo de geração de soluções não é apenas uma caminhada aleatória não direcionada, os AGs exploram informações históricas para encontrar novos pontos de busca onde são esperados melhores desempenhos. Isto é feito através de processos iterativos, onde cada iteração é chamada de geração.

Durante cada iteração, os princípios de seleção e reprodução, os operadores genéticos, são aplicados a uma população de candidatos.

O objetivo dos operadores genéticos é transformar a população através de sucessivas gerações, buscando melhorar a aptidão dos indivíduos. Os operadores genéticos são necessários para que a população se diversifique e mantenha as características de adaptação adquiridas pelas gerações anteriores. Na maior parte dos casos, os AGs utilizam três operadores: seleção, cruzamento e mutação

### **-Seleção**

Este operador genético, também chamado reprodução, seleciona os indivíduos que sofrerão cruzamento e mutação. Da mesma forma que ocorre no processo de

seleção natural, os indivíduos mais qualificados, de acordo com a equação de mérito, têm mais chances de serem escolhidos.

Com os pares formados, passa-se aos demais operadores genéticos: o cruzamento e a mutação.

### **-Cruzamento**

O objetivo do cruzamento é a permutação de material genético entre os pares de indivíduos previamente selecionados. O cruzamento é o operador responsável pela recombinação de características dos pais durante a reprodução, permitindo que as próximas gerações herdem essas características.

### **-Mutaçã**

Entende-se por mutação a inserção de material genético novo na população. Este processo pode ou não ocorrer, de acordo com uma dada probabilidade de mutação. Esta perturbação no código genético dos indivíduos é uma arma poderosa para evita que o algoritmo fique preso em ótimos locais .

De forma resumida as principais definições relacionada aos Algoritmos Genéticos são:

–**Indivíduo**: Cadeia de caracteres representando alguma informação relativa às variáveis do problema. É uma solução potencial para o problema;

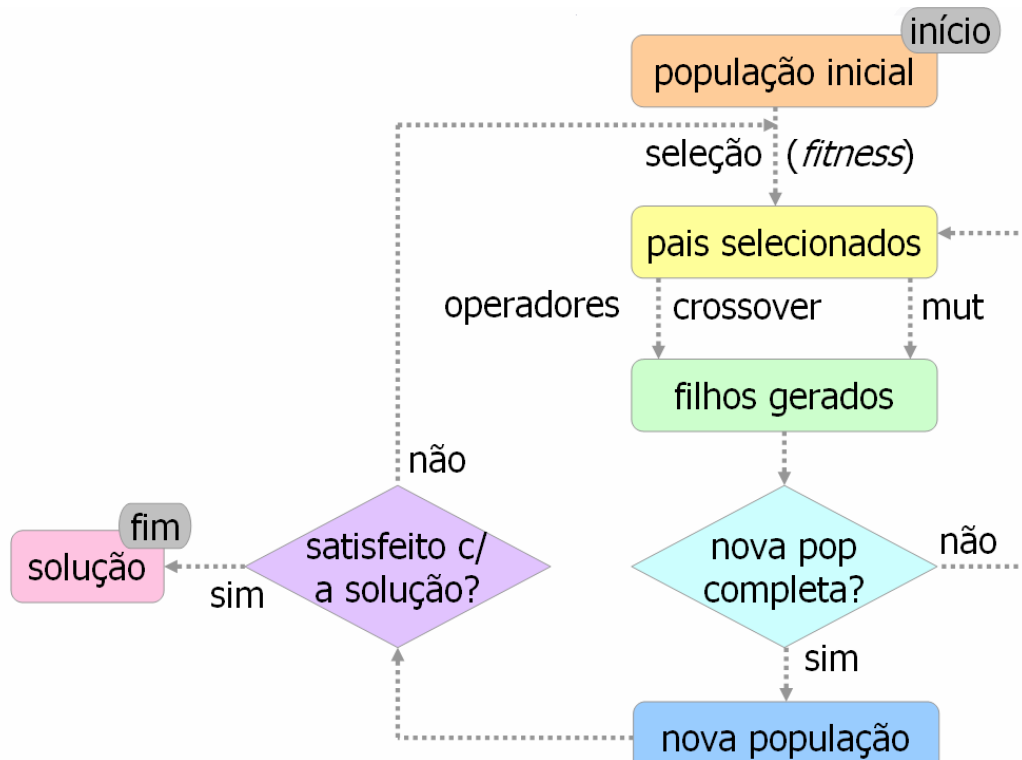
–**Gene**: É a unidade básica do indivíduo. Cada indivíduo tem um certo número de genes, cada um descrevendo uma certa variável do problema;

– **População**: Conjunto de indivíduos ou soluções;

– **Geração**: O número da iteração que o Algoritmo Genético executa;

– **Operações Genéticas:** Operações que o Algoritmo Genético realiza sobre cada um dos indivíduos.

A estrutura do funcionamento de um Algoritmo Genético está apresentada na figura 6 abaixo:



**Figura 6:** Funcionamento de um Algoritmo Genético

Pela Figura 6 o processo de funcionamento de um Algoritmo Genético segue passos simples. O primeiro passo é determinar a população (soluções) inicial para o problema. Esta escolha pode ser feita de forma aleatória ou utilizando alguma heurística.

Com a população inicial definida são selecionados os indivíduos (soluções) mais aptos como pais da nova população por meio do fitness. Escolhidos os indivíduos são

aplicados os operadores de crossover e mutação para gerar novas soluções até que se tenha completa uma nova população.

Este processo é realizado até ser encontrada a solução para o problema

### **3.1-Parâmetros do Algoritmo**

Vários fatores podem alterar a performance de um algoritmo de otimização. Dentre os diversos fatores que podem influenciar a performance dos Algoritmos Genéticos três parâmetros devem ser vistos com um cuidado especial: o tamanho da população, a população inicial e a taxa de mutação.

#### **-Tamanho da População e População Inicial**

A população inicial de indivíduos é, na grande parte das vezes, realizada de forma aleatória. Isto é feito embora existam ocasiões onde é mais apropriado introduzir logo de início, um ou mais indivíduos "interessantes", contendo algum tipo de informação prévia.

A preocupação nesta fase deve ser quanto ao tamanho da população inicial, para que contenha um número de indivíduos com características suficientemente variadas. Segundo Bauer (1994) não é necessário conhecimento prévio de possíveis soluções para a utilização dos Algoritmos Genéticos.

Com uma população pequena o desempenho do algoritmo pode ser prejudicado, pois deste modo a população fornece uma pequena cobertura do espaço de busca do problema. Por outro lado, uma grande população geralmente fornece uma cobertura representativa do domínio do problema, além de prevenir convergências prematuras

para soluções locais ao invés de globais. O principal impacto do parâmetro está relacionada com o desempenho global e a eficiência dos Algoritmos Genéticos.

No entanto, para se trabalhar com grandes populações são necessários maiores recursos computacionais ou que o algoritmo trabalhe por um período de tempo maior. A principal idéia é que quanto maior for a *string* maior deverá ser o tamanho da população para obter uma boa diversidade.

### **-Taxa de Mutação**

A taxa de mutação é utilizada para fornecer novas informações, prevenindo que o resultado tenha uma convergência prematura em um ótimo local. Com isso há um aumento na diversidade populacional e uma maior varredura do espaço de busca.

Apesar de Goldberg (1989) afirmar que o operador de mutação tem um papel secundário em um algoritmo genético simples, é necessário muito cuidado, pois a escolha de uma taxa de mutação muito alta pode fazer com que não ocorra a convergência do algoritmo. Segundo Bauer (1994) a maioria das taxas utilizada varia entre 0.001 e 0.1.

Dentro dos Algoritmos Genéticos existe uma classe específica de algoritmos que lidam com múltiplos objetivos simultaneamente, os Algoritmos Genéticos multiobjetivos, que são tratados na seção seguinte.

## **3.2-Algoritmos Genéticos Multiobjetivos**

Os Algoritmos Genéticos foram criados para lidar com apenas um objetivo, o *fitness*. Em muitos casos temos vários objetivos a serem considerados na otimização, e os mesmos podem não ser agregados em um único objetivo.

Ao se incluir múltiplas medidas de desempenho para uma solução (i.e., múltiplos objetivos), surge um problema para que seja possível comparar de maneira adequada duas soluções distintas.

Goldberg (1989) propôs um ordenamento de soluções baseado no conceito de dominância, onde o *fitness* de uma solução é proporcional ao número de soluções que ela domina.

Segundo Castro (2001) duas são as finalidades quando se deseja determinar o conjunto de Pareto de problemas multiobjetivos via métodos evolucionários:

- 1- Guiar a busca na direção da região ou conjunto ótimo de Pareto;
- 2- Manter a diversidade da população na fronteira de Pareto.

Diferentes versões de algoritmos de otimização multiobjetivos foram baseados nas idéias apresentadas por Goldberg (1989), dentre eles, serão utilizados neste trabalho os seguintes algoritmos genéticos denominados de MOGA e NSGA II, conforme apresentados a seguir:

### **3.2.1- MOGA (*Multi Objective Genetic Algorithm*)**

Desenvolvido em 1993 por Flemming e Fonseca (*apud* Ticona, 2003), este algoritmo se diferencia dos Algoritmos Genéticos tradicionais pela forma que calcula o *ranking* das soluções. O *ranking* de cada solução é calculado da seguinte forma:

$$r_i = 1 + n_i,$$

onde cada solução tem o *ranking* de 1 acrescido do número de soluções que a dominam. Uma solução não dominada tem *ranking* igual a um, ou seja, quanto menor o *ranking* melhor a solução.



Toda a população é verificada e todas as soluções não-dominadas recebem uma posição ou ordem 1. As outras soluções são posicionadas segundo a não dominância delas em relação ao restante da população do seguinte modo: para cada indivíduo (solução), o número de soluções que o dominam estritamente é primeiramente determinado na população, logo, a posição no ordenamento deste indivíduo será este número mais 1.

Após o cálculo do *ranking* as soluções são ordenadas conforme o  $r_i$  obtido. Feita a ordenação é dado a estas soluções um *fitness* preliminar (*raw fitness*,  $raw_i$ ) usando algum tipo de função de classificação. Realizados estes cálculos, é quantificado o valor médio das aptidões para cada ranking da seguinte maneira:

$$F_i = \frac{\sum raw_i}{\mu(r_i)}$$

onde  $\mu(r_i)$  é o número de soluções no *ranking*  $r_i$

No final deste ordenamento poderão existir muitos indivíduos compartilhando a mesma posição no ordenamento. A rotina de seleção usa este ordenamento para selecionar ou remover blocos de pontos até escolher os indivíduos para reprodução. A implementação faz uso do método de formação de nichos para distribuir a população através da região ótima de Pareto, além de compartilhar os valores da função de aptidão.

Para manter a diversidade das soluções é utilizado o *fitness sharing*. O objetivo do mesmo é distribuir as soluções em diferentes espaços de busca. Para cada solução é calculado um valor para o contador de nicho  $nc_i$  usando a expressão abaixo:

$$nc_i = \sum_{j=1}^{\mu(r_i)} Sh(D_{ij})$$

onde  $D_{ij}$  representa a distancia entre duas soluções  $i$  e  $j$  que possuem o mesmo ranking  $r_i$ .

Essa distância é calculada através de:

$$D_{ij} = \sqrt{\sum_{k=1}^m \left( \frac{f_k^{(i)} - f_k^{(j)}}{f_k^{\max} - f_k^{\min}} \right)^2}$$

onde  $f_k^{\max}$  e  $f_k^{\min}$  são os valores máximo e mínimo para a  $k$ -ésima função objetivo.

A função  $Sh$  é conhecida como função de compartilhamento de objetivo, sendo definida como:

$$Sh(d) = 1 - \left( \frac{D}{\sigma share} \right)^\alpha$$

O parâmetro  $D$  é a distância entre duas soluções,  $\alpha$  define o comportamento da função  $Sh$  e  $\sigma$  é chamado raio de nicho, que define a vizinhança de uma solução. O valor de  $\sigma$  é calculado dinamicamente.

E com isso o valor da aptidão compartilhada será:

$$F'_i = \frac{F_i}{nc_i}$$

De acordo com Castro (2001) embora esta estratégia mantenha a diversidade nos valores da função de aptidão, pode não manter a diversidade no conjunto das variáveis, assim, o MOGA pode não estar apto a achar as múltiplas soluções em problemas onde diferentes pontos ótimos de Pareto correspondem para os mesmos valores de aptidões.

Após esses cálculos são utilizados os operadores comuns dos Algoritmos Genéticos: seleção, cruzamento e mutação.

O destaque relevante deste algoritmo é a introdução do ordenamento dos indivíduos por critérios de dominância.

### **3.2.2-NSGA II (Non Dominated Sorting Genetic Algorithm)**

Desenvolvido por Horn em 1994 (*apud* Ticona, 2003), a idéia principal deste algoritmo é a ordenação por elitismo, ou seja, os melhores indivíduos de uma geração serão necessariamente selecionados para a próxima geração (preservando as melhores soluções encontradas até o momento). Outro conceito utilizado neste algoritmo é a distância da multidão (*crowding distance*).

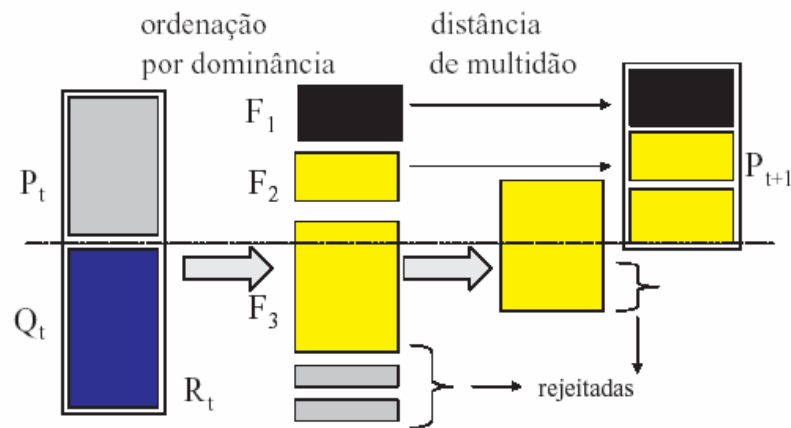
Além da utilização de um procedimento de seleção por ordenamento também é utilizado conjuntamente um método voltado para a criação de nichos para manter a diversidade da população.

Segundo Castro (2001) a diferença desta implementação em relação a um algoritmo genético simples está apenas no modo com que o operador de seleção é empregado. Tanto o operador de recombinação quanto o operador de mutação são os usuais da técnica.

Antes do procedimento de seleção ser aplicado, a população é ordenada com base na não-dominância dos indivíduos, isto é, todos os indivíduos não-dominados da população recebem valores altos de aptidão. Esta aptidão é a mesma para todos os indivíduos na mesma faixa de dominância.

Os melhores indivíduos, os não dominados, irão necessariamente para a próxima geração, iteração do algoritmo. Já os indivíduos na segunda faixa de não dominância em diante são alocados conforme a necessidade de novos pais para a próxima geração.

Na Figura 7 está uma representação gráfica do funcionamento do algoritmo, conforme Ticona (2003):



**Figura 7** NSGA II (DEB, 2001)

A partir da geração inicial composta por  $P+Q$  são selecionadas primeiramente as soluções não dominadas  $F_1$ , depois as soluções na fronteira  $F_2$  (dominadas por uma solução). Estas novas soluções são ordenadas pela dominância e as melhores são passadas para a próxima geração. A diversidade das soluções é enfatizada ao utilizar, como critério de desempate entre as soluções, a distância de multidão.

Esta distância da multidão serve para manter a diversidade na população das soluções não-dominadas, onde as mesmas compartilham os seus valores de aptidão segundo suas distâncias Euclidianas.

Pela figura 7 fica clara a idéia do algoritmo, onde as melhores soluções são levadas para a próxima geração e a partir delas são geradas novas soluções, sempre preservando as melhores soluções encontradas.

De acordo com Castro (2001) a característica mais importante deste algoritmo é que praticamente qualquer número de objetivos pode ser usado para os dois tipos de problemas: maximização ou minimização, bastando mudar o modo como os indivíduos não dominados são identificados.

Com estes conceitos estabelecidos é necessário definir quais o método de pesquisa para avaliar a otimização de carteiras com os Algoritmos Genéticos.

## 4-MÉTODO DE PESQUISA

A proposta deste trabalho é avaliar a otimização de carteiras de investimento com Algoritmos Genéticos levando em consideração duas restrições do mundo real: a existência de lotes de compra e de custos de transação (os custos de transação são considerados os custos de corretagem decorrentes da compra e venda de ações).

O desenvolvimento analítico de métodos de otimização considerando estas duas restrições é impraticável, por este motivo serão utilizados os Algoritmos Genéticos, que são um método de busca extremamente eficaz para lidar com problemas de alta complexidade.

Para tanto serão considerados neste trabalho três Algoritmos Genéticos: o Algoritmo Genético Simples, o MOGA e o NSGA II.

Os ativos disponíveis para investimento serão as ações do índice de ações IBrX-50, composto em 05/07/2006, que são as 50 ações mais líquidas negociadas na Bovespa, conforme apresentado na tabela 2:

Código	Ação	Código	Ação
ACES4	ACESITA	ITSA4	ITAUSA
AMBV4	AMBEV	KLBN4	KLABIN S/A
ARCZ6	ARACRUZ	LIGT3	LIGHT S/A
ARCE3	ARCELOR BR	LAME4	LOJAS AMERIC
BBDC4	BRADESCO	NATU3	NATURA
BRAP4	BRADESPAR	NETC4	NET
BBAS3	BRASIL	PCAR4	P.ACUCAR-CBD
B RTP3	BRASIL T PAR	PRGA3	PERDIGAO S/A
B RTP4	BRASIL T PAR	PETR3	PETROBRAS
B RTO4	BRASIL TELEC	PETR4	PETROBRAS
BRKM5	BRASKEM	SBSP3	SABESP
CCRO3	CCR	SDIA4	SADIA S/A
	RODOVIAS		
CLSC6	CELESC	CSNA3	SID NACIONAL
CMIG4	CEMIG	CRUZ3	SOUZA CRUZ
CTAX3	CONTAX	TNLP3	TELEMAR
CTAX4	CONTAX	TNLP4	TELEMAR
CPL6	COPEL	TMAR5	TELEMAR N L
ELET3	ELETROBRAS	TMCP4	TELEMIG PART
ELET6	ELETROBRAS	TCSL4	TIM PART S/A
EMBR3	EMBRAER	UBBR11	UNIBANCO
EBTP4	EMBRATEL PAR	USIM5	USIMINAS
GGBR4	GERDAU	VCPA4	V C P
GOAU4	GERDAU MET	VALE3	VALE R DOCE
GOLL4	GOL	VALE5	VALE R DOCE
PTIP4	IPIRANGA PET	VIVO4	VIVO
ITAU4	ITAUBANCO		

**Tabela 2:** Ações do IBrX-50 (Fonte: Bovespa)

#### 4.1-Representação do Problema

O primeiro passo para a utilização dos Algoritmos Genéticos é a “transcrição” do problema para o formato de uma *string*, ou cromossomo, que é maneira que o algoritmo trabalha com as soluções.

Cada solução potencial terá a representação em forma de *string* da seguinte maneira:

Ação	A	B	C	...	Z
Solução 1	100	0	25	...	3
Solução 2	70	10	6	...	1

Na Solução 1 é proposta a compra de 100 lotes da ação A, nenhum lote da ação B, 25 lotes da ação C e assim sucessivamente. Já a solução 2 propõe a compra de 70 lotes da ação A, 10 da ação B, 6 lotes da ação C e assim sucessivamente. Ou seja, cada valor representa o lote de ações a ser comprado.

#### **4,2-Cruzamento e Mutaç o**

Ap s a seleç o das melhores soluç es   feito o cruzamento para geraç o de novas soluç es, ou filhos.

<b>Aç�o</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>...</b>	<b>Z</b>
<b>Pai 1</b>	100	0	25	...	3
<b>Pai 2</b>	70	10	6	...	1

Com o cruzamento   feita a troca na composiç o das soluç es no ponto de corte indicado, fazendo com que as novas soluç es sejam dadas por:

<b>Aç�o</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>...</b>	<b>Z</b>
<b>Filho 1</b>	100	0	6	...	1
<b>Filho 2</b>	70	10	5	...	3

O ponto de corte utilizado neste trabalho ser , considerando n a es dispon veis,  $n / 2$ . Com este ponto de corte as novas soluç es ser o criadas a partir da mudanç a de metade de sua estrutura.

A decis o quanto ao ponto de corte foi feita por conveni ncia e o impacto de poss veis mudanç as no ponto de corte sobre o resultado do algoritmo est o al m do escopo deste trabalho.

Feito o cruzamento, o operador genético a ser aplicado é a mutação, que evita com que as soluções fiquem presas em “ótimos locais”. Para cada ação de cada solução é jogada uma “moeda”, sendo que a probabilidade de sucesso é a taxa de mutação. Ocorrendo o sucesso é feita uma alteração na quantidade investida na ação, sendo a adição de um lote à quantidade investida na ação.

As taxa de mutação consideradas neste trabalho serão: 1%, 5%, 10%, 15%, 20%, 25% e 30%;

### ***4.3-Quantidade de Ativos Disponíveis***

Para analisar a performance dos Algoritmos Genéticos no problema de otimização de carteiras foi escolhido trabalhar com quatro problemas distintos, cada um sendo a otimização de uma carteira com uma quantidade diferente de ativos disponíveis.

A escolha dos ativos disponíveis para cada problema foi escolhida da seguinte forma aleatória:

- 1- Ordenação em ordem alfabética dos 50 ativos disponíveis;
- 2- Para o problema de 5 ações escolher as 5 primeiras ações;
- 3- Para o problema de 10 ações escolher da 6<sup>a</sup> à 16<sup>a</sup> ação;
- 4- Para o problema de 25 ações escolher da 17<sup>a</sup> à 42<sup>a</sup> ação.

Com isto, os ativos a serem trabalhados em cada um dos problemas serão:



Quant	Ações		
<b>5</b>	ACES		
	AMBV		
	ARCZ		
	ARCE		
	BBDC		
<b>10</b>	BRAP		
	BBAS		
	BRTP		
	BRTO		
	BRKM		
	CCRO		
	CLSC		
	CMIG		
	CTAX		
	CPLA		
	<b>25</b>	ELET	PETR
		EMBR	SBSP
		EBTP	SDIA
		GGBR	CSNA
		GOAU	CRUZ
ITSA		TNLP	
KLBN		TMAR	
LIGT		TMCP	
LAME		TCSL	
NATU		UBBR	
NETC		USIM	
PCAR		VCPA	
PRGA			

**Tabela 3:** Ações para cada problema

#### **4.4-População Inicial**

Para lidar com o número de variáveis aleatórias do problema, a quantidade investida em cada ativo disponível (50 ações), foi escolhido trabalhar com uma população de 100 soluções.

Primeiramente a população inicial de soluções foi escolhida de forma aleatória. Com esta escolha da população inicial o tempo estimado para chegar ao resultado final, considerando o algoritmo genético simples, de um período (um dia) era de cerca de 17 horas.

Com um tempo de processamento elevado desta maneira se tornou imperativo o desenvolvimento de uma heurística para seleção da população inicial de maneira que o tempo de processamento fosse reduzido.

Para a escolha da população inicial foi desenvolvida a seguinte heurística:

- 1- Primeiramente é efetuado o cálculo das soluções na Fronteira de Pareto pelo método de Markowitz, sem considerar nenhuma restrição de lotes mínimos ou custos de transação;
- 2- É realizada a escolha das  $n$  primeiras soluções para a população inicial de tamanho  $n$  (no caso deste trabalho  $n=100$ );
- 3- Para cada solução  $n$  é feito o arredondamento do valor investido na ação  $i$  para o maior lote de compra da ação possível.

Com as soluções iniciais calculadas com esta heurística o tempo de processamento de um período de um dia passou de 17 horas para cerca de 5 minutos.

#### **4.5-Custos de Transação**

Na movimentação financeira de compra e venda de ativos existem, no mundo real, custos para realizar tais operações. Esta característica, os custos de transação, é ignorada no modelo desenvolvido por Markowitz de maneira a facilitar o desenvolvimento analítico do modelo, mas será considerada neste trabalho.

Como custos de transação foram escolhidos, por conveniência, os custos de corretagem disponibilizados no site da corretora Hedging-Griffo. O custo de corretagem é formado por um valor fixo somado a um valor variável de acordo com o volume total das operações realizadas no mesmo dia, conforme a tabela a seguir:

<b>Valor da operação</b>	<b>Taxa</b>	<b>Custo Fixo</b>
Até R\$ 135,05	0,00%	R\$ 2,70
De R\$ 135,06 até R\$ 498,615	2,00%	R\$ 0,00
De R\$ 498,62 até R\$ 1.514,68	1,50%	R\$ 2,49
De R\$ 1.514,69 até R\$ 3.029,37	1,00%	R\$ 10,06
A partir de R\$ 3.029,38	0,50%	R\$ 25,21

**Tabela 4:** Custos de Transação (Fonte: Hedging-Griffo)

Como custos de transação foram consideradas as mudanças nos totais investidos em cada ação. Por exemplo:

<b>Ação</b>	<b>Período 1</b>	<b>Período 2</b>	<b>Varição</b>
<b>A</b>	100	70	30
<b>B</b>	10	0	10
<b>C</b>	0	20	20
<b>Total</b>			<b>60</b>

**Tabela 5:** Mudança de investimentos entre períodos

No período 1 a carteira continha R\$100 da ação A e no período 2 passa a conter R\$70 o montante a ser utilizado como base para o cálculo do custo de corretagem, custo de transação, será de R\$30.

No exemplo acima, o total a ser considerado como base para o cálculo dos custos de transação será de R\$ 60, ou seja, são consideradas as variações tanto na hora de comprar como vender ações entre os períodos.

### 4.3-Lotes de Compra

Outra característica do mercado financeiro é que os ativos são transacionados em múltiplos de quantidades mínimas, conhecidos como lotes. Na Bovespa os lotes mínimos de negociação e a respectiva cotação de cada ativo considerado neste trabalho são apresentados na tabela 4:

Ação	Lote	Cotação	Ação	Lote	Cotação
ACES	100	unitária	KLBN	1.000	unitária
AMBV	10.000	por lote de mil ações	LIGT	10000	por lote de mil ações
ARCZ	100	unitária	LAME	100.000	por lote de mil ações
ARCE	100	unitária	NATU	100	unitária
BBDC	100	unitária	NETC	100	unitária
BRAP	100	unitária	PCAR	10000	por lote de mil ações
BBAS	100	unitária	PRGA	100	unitária
B RTP	100.000	por lote de mil ações	PETR	100	unitária
BRT0	100.000	por lote de mil ações	SBSP	10.000	por lote de mil ações
BRKM	100	unitária	SDIA	1.000	unitária
CCRO	100	unitária	CSNA	100	unitária
CLSC	1.000	unitária	CRUZ	100	unitária
CMIG	100.000	por lote de mil ações	TNLP	100	unitária
CTAX	100	unitária	TMAR	100	unitária
CPL E	100.000	por lote de mil ações	TMCP	100.000	por lote de mil ações
ELET	100.000	por lote de mil ações	TCSL	100.000	por lote de mil ações
EMBR	100	unitária	UBBR	100	unitária
EBTP	100.000	por lote de mil ações	USIM	100	unitária
GGBR	100	unitária	VCPA	100	unitária
GOAU	100	unitária	VALE	100	unitária
ITSA	1.000	unitária	VIVO	100	unitária

Tabela 6: Lotes de Transação (Fonte: Bovespa)

#### **4.4- Previsão dos Retornos, Volatilidades e Covariâncias**

Além dos custos de transação e dos lotes mínimos, é necessário definir qual a informação a ser considerada para a composição da carteira. A inserção apenas da série histórica dos preços das ações é de poucas possibilidades se comparada à simulação de diferentes preços das ações.

A simulação considerada neste trabalho é baseada no horizonte de um dia, ou seja, são feitas as previsões para um dia dos preços das ações e, com os preços definidos através da simulação, é definida a composição da carteira. Como critério para gerar as previsões será utilizado neste trabalho o modelo IGARCH.

O modelo IGARCH parte de uma característica importante das séries financeiras que, enquanto os retornos são independentes, o quadrado dos retornos não o são. Com esta autocorrelação do quadrado dos retornos, fica inválida a hipótese de heterocedasticidade (a heterocedasticidade implica que a média e a variância de uma série temporal são constantes ao longo do tempo).

Em 1982 Engle desenvolveu o chamado Modelo Heterocedasticidade Auto Regressiva (ARCH) que leva em consideração o fato que a variância do erro atual é uma função das variância dos erros passados. O modelo ARCH relaciona a variância do erro ao quadrado dos erros dos períodos anteriores, modelando desta maneira séries que apresentem sua volatilidade variando no tempo. Especificamente, considerando  $\epsilon_t$  os retornos e assumindo que  $\epsilon_t = \sigma_t z_t$ , onde  $z_t \sim N(0,1)$ , a série  $\sigma^2$  é modelada por:

$$\sigma_t^2 = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \dots + \alpha_p \epsilon_{t-p}^2$$

Onde  $\alpha_0 > 0$  e  $\alpha_i \geq 0, i > 0$

Em 1986 Bollerslev desenvolveu o modelo GARCH, generalizando o modelo ARCH, onde a variância dos erros segue um processo auto regressivo e de médias móveis (ARMA). Neste caso, um modelo GARCH(p,q) é dado por:

$$\sigma_t^2 = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \dots + \alpha_p \epsilon_{t-p}^2 + \beta_1 \sigma_{t-1}^2 + \dots + \beta_q \sigma_{t-q}^2$$

Se a soma dos coeficientes alfa e beta do modelo for igual a 1 o modelo é chamado IGARCH (GARCH Integrado). A grande vantagem destes modelos é levar em consideração que os períodos de alta volatilidade são “agrupados” e que essa volatilidade segue um processo conhecido.

Um caso específico do IGARCH é a média móvel alisada exponencialmente usado pelo Riskmetrics (1996) que será utilizada para a previsão de retornos, variâncias e correlações neste trabalho.

A variância seguirá o seguinte processo:

$$\sigma_t^2 = \lambda \sigma_{t-1}^2 + (1 - \lambda) r_{t-1}^2$$

onde,  $\sigma_t$  é a variância no tempo t,  $r_t$  é o retorno no período t e  $\lambda$  é denominado fator de decaimento. Segundo a metodologia do Riskmetrics (1996) será utilizado como fator de decaimento 0,94 para dados diários.

Da mesma forma a covariância será dada por:

$$Cov(a,b)_t = \lambda Cov(a,b)_{t-1} + (1 - \lambda) r(a)_{t-1} r(b)_{t-1}$$

Já os preços, e conseqüentemente os retornos, serão dados por:

$$p_t = p_{t-1} + \sigma_t \epsilon_t, \text{ onde } \epsilon_t \sim N(0,1)$$

## 5-ANÁLISE DOS RESULTADOS

Além da quantidade de ativos disponíveis, fatores que influenciam o desempenho do algoritmo como a taxa de mutação e o algoritmo genético também foram considerados. Cada uma destas variáveis pode ser considerada um fator a ser testado, e para cada fator serão considerados os seguintes níveis:

- Ativos Disponíveis: 5, 10, 25 e 50;
- Taxa de mutação (em %): 1, 5, 10, 15, 20, 25 e 30;
- Algoritmo utilizado: Simples, MOGA e NPGA.

Considerando que este trabalho contemplará todas as combinações destes fatores, um experimento fatorial completo, serão necessários:

$$4 \times 7 \times 3 = 84 \text{ testes.}$$

Os resultados serão gerados em duas etapas a partir do seguinte processo:

- Primeiramente a carteira é otimizada com as informações disponíveis, a previsão de retorno e as covariâncias para um dia adiante, pelo IGARCH;
- Após a otimização é calculado, no dia seguinte, o retorno realizado com os dados de mercado.

Para os resultados dos algoritmos não serem influenciados pelo desempenho particular de um dia, foi escolhido trabalhar com os resultados de 90 dias e, a partir deles, ter uma média do desempenho, e a respectiva variância, para cada combinação de fatores.

Está disponível inicialmente R\$ 100 mil para a composição da carteira, não é permitida a venda a descoberto, a venda de uma ação que o investidor não possua, os possíveis ganhos ou perdas são repassados para o período seguinte.

Outro fator a ser considerado é o fato dos Algoritmos Genéticos Multiobjetivo não apresentarem nenhum problema em lidar com dois objetivos conflitantes como o risco e o retorno. Já com o algoritmo genético simples isto não ocorre, sendo necessário agregar estes dois objetivos conflitantes em uma só métrica.

Foi utilizada a variável Retorno Médio / Desvio Padrão como maneira de agregar os dois objetivos conflitantes. Com esta variável tem-se que quanto maior o Retorno Médio melhor o índice, quando menor o Desvio Padrão, melhor o índice, ou seja, é desejado Retorno maior e Desvio Padrão menor.

Os resultados para cada combinação de fatores estão na tabela abaixo:

		Simples			MOGA			NSGA			
		Mutação(%)	Retorno	DesvPad	Tempo	Retorno	DesvPad	Tempo	Retorno	DesvPad	Tempo
5 ações	1	0,7422	12,07	3325	0,934	17,02	6229	1,032	21,077	6790	
	5	0,9872	20,33	3291	0,996	20,20	6445	1,0093	21,0231	6612	
	10	0,0713	3,54	3309	0,231	9,17	7766	1,006	19,078	9709	
	15	0,6785	14,51	3315	0,660	14,53	7860	0,5032	9,0733	9687	
	20	1,0223	23,07	3320	0,982	16,91	6452	1,142	23,1005	11994	
	25	1,0032	22,11	3330	0,735	15,02	4044	0,8704	11,0062	6830	
10 ações	30	0,7723	18,53	3324	0,103	8,05	8464	1,008	19,126	12004	
	1	0,6172	21,13	9916	1,4021	78,346	18687	1,003	55,0561	20369	
	5	1,031	42,51	9612	0,3628	16,0459	19334	0,1431	10,273	19836	
	10	1,006	34,12	9744	1,012	27,913	23299	1,0502	33,146	29128	
	15	0,8103	26,36	9699	1,0214	32,018	23581	0,9031	20,103	29061	
	20	0,5127	19,21	9441	0,3766	13,219	19355	1,105	31,794	35983	
25 ações	25	1,093	29,02	9625	0,8168	11,0564	12132	1,3016	45,7032	20489	
	30	0,987	13,21	9387	1,0953	25,1504	25391	1,206	68,071	36011	
	1	0,8647	9,13	25320	-0,1482	8,314	52172	1,039	66,703	56869	
	5	0,0941	2,07	24427	-1,936	33,695	50638	0,0402	1,753	51951	
	10	1,3129	46,33	22307	1,063	54,018	61020	1,081	36,7901	76288	
	15	1,903	89,21	23062	0,7082	17,694	61760	2,021	76,65	76113	
50 ações	20	0,9125	16,13	22663	1,101	54,028	50692	1,022	36,296	94240	
	25	1,0179	27,06	24813	0,5791	12,013	31774	0,8731	10,038	53661	
	30	0,3901	32,12	24582	0,4424	23,017	66500	2,762	73,9901	94314	
	1	-0,1147	20,61	49881	-1,9428	18,1154	102778	1,443	75,0578	112032	
	5	1,4441	22,55	51297	-1,3981	63,4957	106339	-0,0841	61,6234	109098	
	10	0,1766	33,72	46845	-0,9109	52,6416	128142	1,5212	53,7456	160205	
50 ações	15	1,6119	36,57	48430	1,5294	52,0029	129696	0,8891	23,783	159838	
	20	0,4038	33,04	47593	0,4626	33,018	106453	3,195	73,8908	197905	
	25	2,6139	39,12	52108	0,6188	57,5937	66725	2,7157	44,6503	112689	
	30	0,2787	43,28	51622	0,2533	55,5447	139650	2,186	76,829	198060	

Tabela 7: Resultados

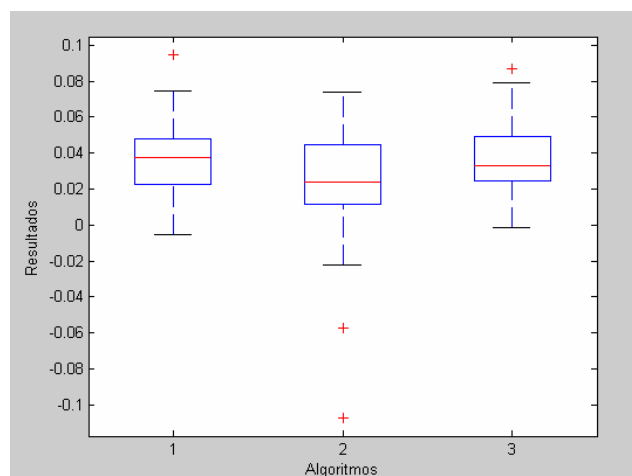


Na utilização do Algoritmo Genético Simples, apenas em um caso o Retorno Médio foi negativo, no problema com 50 ações e taxa de mutação de 1%, onde o Retorno Médio foi de -0,1147. O mesmo ocorre com o NSGA, onde apenas um caso apresenta Retorno Médio negativo, o problema com 50 ações e taxa de mutação de 5%, com -0,0841.

O menor Desvio Padrão, de 3,54, ocorre no Algoritmo Genético Simples, no problema com 5 ações e taxa de mutação de 10%. O maior Desvio Padrão, de 89,21, também ocorre no Algoritmo Genético Simples, no problema de 25 ações com taxa de mutação de 15%.

A primeira questão a ser verificada é se algum algoritmo apresenta um desempenho melhor.

Como critério de análise também foi utilizada a variável Retorno Médio / Desvio Padrão como maneira de agregar os resultados dos algoritmos. Pela Figura 8, da distribuição dos resultados, há uma indicação que o comportamento do resultado do Algoritmo Genético Simples (1) é tão bom quanto o desempenho do Algoritmo NSGA II (3).

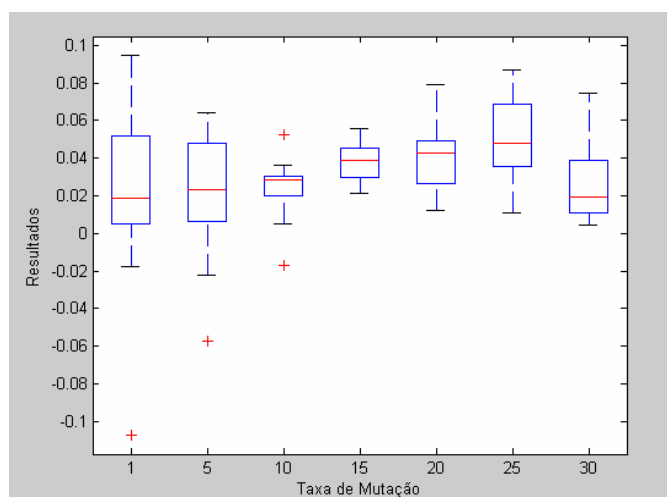


**Figura 8:** Distribuição dos Resultados por Algoritmo

Enquanto o MOGA apresenta alguns resultados negativos, o Algoritmo Genético Simples e o NSGA apresentam um caso cada. E estes resultados negativos do MOGA não são explicados pela dispersão dos dados, dado que não ocorrem resultados melhores que os do Algoritmo Genético Simples e do NSGA.

A segunda questão a ser respondida é se existe alguma taxa de mutação que apresente melhor performance para o problema de otimização de carteiras.

Isto pode ser analisado pela dispersão dos resultados por cada taxa de mutação, conforme pode ser observado pela Figura 9.



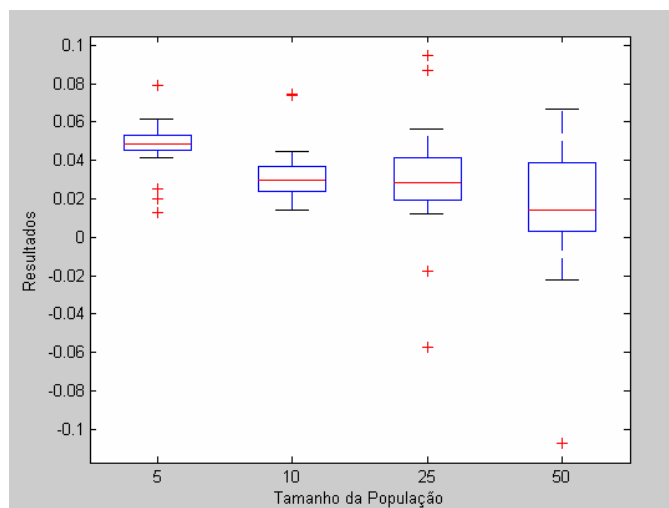
**Figura 9:** Distribuição dos Resultados por Taxa de Mutação

Pela dispersão dos resultados há um indicativo que o desempenho do algoritmo é melhor com uma taxa ao redor de 15%, onde a dispersão dos dados é pequena e os resultados concentrados na faixa de 0,04.

A taxa de mutação com resultados mais instáveis, isto é mais dispersos, é 1% seguida por 5%. Pelo gráfico há uma evidência que os resultados vão se estabilizando conforme a taxa de mutação aumenta entre 10% e 15%, e depois desta faixa começam a se tornar mais dispersos novamente.

A terceira questão a ser analisada é se a quantidade de ativos disponíveis para investimento tem uma influência direta no desempenho dos algoritmos.

Isto pode ser observado pela Figura 10 :



**Figura 10:** Distribuição dos Resultados por Tamanho de População

Pode-se observar que ocorre o melhor desempenho dos algoritmos conforme o número de ativos disponíveis diminui. Isto pode ser explicado pelo fato que quanto menor o número de ativos disponíveis, menor espaço de busca, fazendo com que o trabalho de busca de boas soluções seja mais fácil. A variabilidade dos resultados é reduzida e o desempenho se torna melhor.

Apesar do problema com 25 ações disponíveis apresentar os dois melhores resultados, isto pode ser explicado pela dispersão dos dados que abrange até o segundo pior resultado.

Além destes indicativos, é necessária uma análise formal dos dados, de forma que se possa afirmar quais variáveis impactam no resultados dos algoritmos, o quanto elas impactam e se existe interação entre os fatores considerados no desempenho dos algoritmos.

O método a ser utilizado para analisar a influência dos fatores nos resultados foi a análise de variância (ANOVA).

A análise de variância é um teste estatístico que visa fundamentalmente verificar se existe uma diferença significativa entre as médias de grupos diferentes e se os fatores exercem influência em alguma variável dependente.

Os fatores propostos podem ser de origem qualitativa ou quantitativa, mas a variável dependente necessariamente deverá ser contínua.

A tabela ANOVA com os resultados para cada fator considerado está apresentada abaixo:

Tabela ANOVA					
Fator	Soma Quad	g.l.	Quad Medio	F	Prob>F
X1	0,00528	2	0,00264	5,75	0,0066
X2	0,0033	4	0,00083	1,8	0,1496
X3	0,00371	2	0,00186	4,04	0,0257
X1*X2	0,0077	12	0,00065	1,41	0,2043
X1*X3	0,00723	6	0,00121	2,62	0,0316
X2*X3	0,00911	16	0,00057	1,24	0,2856
Erro	0,01747	38	0,00046		
Total	0,06791	83			

**Tabela 8:** Tabela ANOVA

Pela tabela Anova fica claro a importância do fator Algoritmo utilizado (X1), dado que o F calculado é maior que um valor crítico, sendo significativa a um nível de 5% de confiança. Isto indica que o dependendo do algoritmo existirão resultados diferentes.

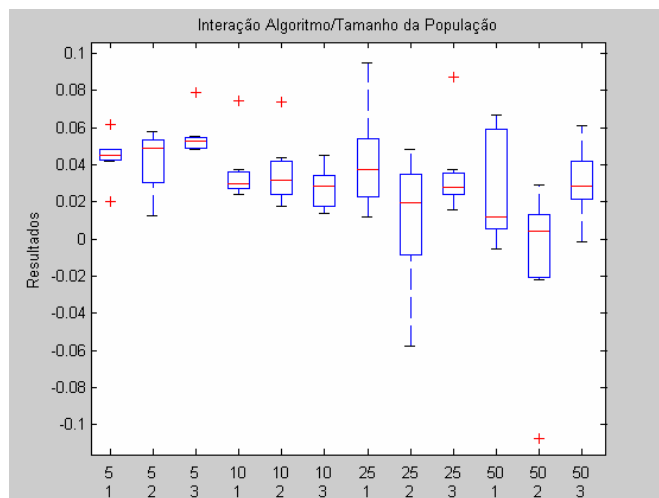
Já o fator Taxa de Mutação (X2) não apresenta efeito significativo no resultado dos algoritmos, não sendo significativa a um nível de 5% de confiança.

O fator Tamanho da População (X3) tem influência nos resultados sendo o fator significativa a 5% de confiança.

Além da influência isolada de cada fator é necessário também analisar a interação dos fatores nos resultados dos algoritmos.

Entre as interações entre os fatores, apenas a interação Algoritmo utilizado e Tamanho da População ( $X1 * X3$ ) é significativa estatisticamente, considerando um nível de confiança de 5%.

As outras possíveis interações não foram significantes estatisticamente, ou seja, a utilização destas interações não explicou a variabilidade dos dados.



**Figura 11:** Distribuição dos Resultados por Algoritmo e Tamanho de População

Pela figura 11 a dispersão dos resultados há uma indicação que o principal fator atuando é o tamanho da população, onde os resultados dos três algoritmos (os três primeiros na figura 11) apresentam um resultado melhor, e a dispersão dos mesmos é bem reduzida quando comparada à dispersão para os outros tamanhos de população.

Conforme é aumentado o número de ações disponíveis os resultados se tornam mais dispersos, e não necessariamente com melhores resultados (o que indicaria que o risco sendo aumentado as oportunidades para maiores retornos também aumentariam) .

## 6-CONSIDERAÇÕES FINAIS

Os Algoritmos Genéticos são uma poderosa ferramenta para busca de soluções em problemas com alto nível de complexidade. Desde os conceitos básicos realizados por Holland, os mesmos vêm sendo utilizados em várias áreas de pesquisa e em situações do mundo real com bons resultados.

O método de otimização de carteiras desenvolvido por Markowitz se valeu de algumas hipóteses simplificadoras que permitiram o desenvolvimento analítico do modelo.

A utilização dos Algoritmos Genéticos para a otimização de carteiras, relaxando as hipótese de divisibilidade dos ativos e ausência de custos operacionais, se mostrou perfeitamente aplicável.

Apesar de ser um problema com dois objetivos conflitantes, risco e retorno, o desempenho dos Algoritmos Genéticos Multiobjetivo não foi superior à performance do algoritmo genético simples, o que demonstra a aplicabilidade da ferramenta em um problema tão complexo.

O retorno diário dos algoritmos ficou em torno de 0,04%, além de poucas ocorrências de retornos negativos, o que demonstra uma boa performance.

Como continuidade deste trabalho fica a proposta de novos estudos utilizando outras fontes de informação além da previsão para um período a frente, inclusão de outros ativos, como derivativos, e a venda a descoberto.

Com relação aos Algoritmos Genéticos fica a proposta de novos estudos para buscar a taxa de mutação ótima e a utilização de diferentes pontos de corte no cruzamento e a respectiva influência de cada um na otimização de carteiras.

## 7-REFERÊNCIAS BIBLIOGRÁFICAS

- ACERBI, C.; SIMONETTI, P.:(2002). *Portfolio Optimization with Spectral Measures of Risk*. Abaxbank Technical Paper.
- AMORIM, E. A. (2006). *Fluxo de Potência Ótimo em Sistemas Multimercados Através de um Algoritmo Genético Multi Objetivo*. Tese de Doutorado, UNESP Ilha Solteira.
- ARAUJO, A. (1984). *Introdução à Economia Matemática*. 14º Colóquio Brasileiro de Matemática. IMPA-RJ.
- ÁVILA, S. (2002). *Algoritmos Genéticos Aplicados na Otimização de Antenas Refletoras*. Dissertação de Mestrado, Universidade Federal de Santa Catarina.
- BAUER JR, RICHARD J. (1994). *Genetic Algorithms and Investment Strategies*, John Wiley & Sons, New York, NY.
- BENSALAH, Y. (2002). *Asset Allocation Using Extreme Value Theory*. Bank of Canada Working Paper.
- BINSBERGEN, J. H.; BRANDT, M. W. (2005). *Optimal Asset Allocation in Asset and Liability Management*. Fuqua Business School. Duke University.
- BRADLEY, B.; TAQQU, M.; (2004). *An Extreme Value Theory Approach To The Allocation of Multiple Assets*. International Journal of Theoretical and Applied Finance, vol 7, n. 8, pg 1031-1068.
- CASTRO, R (2001). *Otimização de Estruturas Multiobjetivos via Algoritmos Genéticos*. Tese de Doutorado, COPPE UFRJ.
- CHEKHLOV, A.; URYASEV,S.; ZABARANKIN, M.:(2005). *Drawdown Measures in Portfolio Optimzation*. International Journal of Theoretical and Applied Finance, vol 8, n. 1, pg 13-58.

- DANTAS, A. L. (2006) *Otimização Multiperíodo por Média-Variância sem Posições a Descoberto em Ativos de Risco*. Dissertação de Mestrado, POLI-USP.
- DEB, K.(2001) *Multi-objective optimization using evolutionary algorithms*. New York: John Wiley and Sons Inc.
- DENTCHEVA, D.; RUSZCZYNSKI, A.(2006). *Portfolio Optimzation with Stochastic Dominance Constraints*. Journal of Banking and Finance, Vol 30, Issue 2, pg 433-451.
- DI CLEMENTE, A.; ROMANO,C. (2003). *Beyond Markowitz: Building Optimal Portfolio Using Non-Elliptical Asset Return Distributions*.White Paper
- DI GIORGI, E. (2002). *A Note on Portfolio Selection Under Various Risk Measures*. Institute for Empirical Research in Economics, Working Paper No. 122. University of Zurich.
- FACCIOLI, R. (2007). *Algoritmo Híbrido Multiobjetivo para Predição de Estrutura Terciária de Proteínas*. Dissertação de Mestrado. EESC-USP.
- FOGEL, DAVID B. (1995). *Evolutionary Computation. Toward a New Philosophy of Machine Intelligence*, IEEE, New York, NY.
- GILLI, M.; KELLEZI, E. (2000). *A Heuristic Approach to Portfolio Optimization*. International Center for Financial Asset Management and Engineering, University of Geneva.
- GOLDBERG, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, Mass.
- JAMES, BARRY. (1996). *Probabilidade : um curso em nível intermediário*. 2a. ed., IMPA-RJ.
- JORION, P. (1999). *Value At Risk: A Nova Fonte de Referência para o Controle de Riscos Financeiros*. 2 ed. BM&F, SP.



- KALIN, D.; ZAGST, R.(1999). *Portfolio Optimization: Volatility Constraints versus Shortfall Constraints*. OR Spektrum, Vol 21, No 1-2. pg 97--122.
- KATO, F. H.; (2004). *Análise de Carteiras em Tempo Discreto*. Dissertação de Mestrado. FEA-USP.
- KONNO, H.; YAMAZAKI, H. (1991). *Mean-Absolute Deviation Portfolio Optimization Model and Its Application to Tokyo Stock Market*, Management Science, Vol. 37, N° 5, pp.519-531
- LAUPRETE, G.J.; SAMAROV, A. M.; WELSCH R. E.(2002) *Robust Portfolio Optimization*. *Metrika*, vol 55, No 1-2, pg 139-149.
- LEE, J.; LEE, S.; CHANG, S.; AHN, B.(2005). *A Comparison of GA and PSO for Excess Return Evaluation in Stock Markets*. Lectures Notes in Computer Science 3562, pg 221-230.
- LUTHI, H. J.; DOEGE, J. (2005). *Convex Risk Measures for Portfolio Optimization and Concepts of Flexibility*. White Paper
- MARKOWITZ, HARRY M. (1952). *Portfolio selection*, Journal of Finance, 7(1), 77-91
- MARINGER, D. (2003). *Distribution Assumption and Risk Constraints in Portfolio Optimization*. Discussion Paper No.: 2003-008E. Erfurt University.
- MARINGER, D.; WINKER, P.(2003). *Portfolio Optimization under Different Risk Constraints with Modified Memetic Algorithms*. Discussion Paper No.: 2003-005E. Erfurt University.
- MARTIN, R.; RACHEV, S.; SIBOULET, F. (2003). *Phi-Alpha Optimal Portfolios And Extreme Risk Management*. Wilmott Magazine of Finance, November.
- MACASTROPA, F. C.; (2006). *A Aplicabilidade da Moderna Teoria de Portifólios em Títulos de Renda Fixa Internacionais*. Dissertação de Mestrado FEA-USP

- MARZANO, L. G. B.(2004). *Otimização de Portfólio de Contratos de Energia em Sistemas Hidrotérmicos com Despacho Centralizado*. Dissertação de Doutorado, PUC-Rio.
- NABHOLZ, R. (2006). *Seleção Ótima de Ativos Multi-Período com Restrições Intermediárias Utilizando o Critério Média-Variância*. Tese de Doutorado, POLI-USP.
- OH, K.Y; KIM, T.Y.;MIN, (2005). *Using Genetic Algorithms to Support Portfolio Optimization for Index Fund*. Management.Expert Systems with Applications. 28, pg 371-379.
- PIRVU, T. A. (2005). *Portfolio Optimization under Value At Risk Constraint*. Technical Report, University of British Columbia.
- RIBEIRO, C.; JÚDICE, J. J.; SANTOS, J.(2003). *Análise Comparativa dos Modelos de Seleção de Carteiras de Ações de Markowitz e Konno*. Relatório Técnico.
- SOUZA, M. J. F.(2007) *Otimização Combinatória*. Notas de Aula, Universidade Federal de Ouro Preto.
- TICONA, W.G.C (2003).*Aplicação de Algoritmos Genéticos Multiobjetivos para Alinhamento de Sequências Biológicas*. Dissertação de Mestrado.ICMC-USP.
- XIA, Y.;BAODING,L.;SHOUYANG,W.; LAI, K.K.; (2000). *A Model for Portfolio Selection with Order of Expected Returns*. Computers & Operations Research, 27, pg 409-422.
- [http://www.griffo.com.br/home\\_fora/custos\\_operacao.asp](http://www.griffo.com.br/home_fora/custos_operacao.asp) acessado no dia 24/07/2006.
- <http://www.bovespa.com.br> acessado no dia 25/07/2006

## ANEXO 1 – Desenvolvimento do Modelo de Markowitz

Considerando uma carteira de  $N$  ativos:

$$R = \sum_{i=1}^N w_i r_i, \quad \sum_{i=1}^N w_i = 1$$

Sendo  $R$  o retorno da carteira,  $r_i$  o retorno do ativo  $i$  e  $w_i$  a proporção investida no ativo  $i$ .

A variância desta carteira será dada por:

$$\begin{aligned} \text{var}[R] &= E\left[\left(R - \bar{R}\right)^2\right] = \\ &= E\left[\left(\sum_{i=1}^N w_i r_i - E(R)\right)^2\right] = \\ &= E\left[\left(\sum_{i=1}^N w_i r_i - \sum_{i=1}^N w_i E[r_i]\right)^2\right] = \\ &= E\left[\left(\sum_{i=1}^N w_i (r_i - E[r_i])\right)^2\right] = \\ &= E\left[\left(\sum_{i=1}^N w_i (r_i - E[r_i])\right)\left(\sum_{j=1}^N w_j (r_j - E[r_j])\right)\right] = \\ &= E\left[\sum_{i=1}^N \sum_{j=1}^N w_i w_j (r_i - E[r_i])(r_j - E[r_j])\right] = \\ &= \sum_{i=1}^N \sum_{j=1}^N w_i w_j E[(r_i - E[r_i])(r_j - E[r_j])] = \\ &= \sum_{i=1}^N \sum_{j=1}^N w_i w_j \text{cov}(r_i, r_j) = \\ &= \sum_{i=1}^N \sum_{j=1}^N w_i w_j \sigma_i \sigma_j \rho_{ij} \end{aligned}$$

Com isso o retorno total da carteira corresponde à soma dos retornos dos ativos, já a variabilidade da carteira é menor que a soma da variabilidade dos ativos.

Conforme for aumentando o número de ativos da carteira, há uma redução na variabilidade total.

Considerando a variância de uma carteira de  $N$  ações:

$$\begin{aligned}\text{var}[R] &= \sum_{i=1}^N \sum_{j=1}^N w_i w_j \text{cov}(r_i, r_j) = \\ &= \sum_{i=1}^N w_i^2 \sigma_i^2 + \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N w_i w_j \text{cov}(r_i, r_j)\end{aligned}$$

Agora consideremos que iremos alocar o mesmo montante em todas as ações, teremos então que a variância do portfólio será:

$$\text{var}[R] = \sum_{i=1}^N \left(\frac{1}{N}\right)_i^2 \sigma_i^2 + \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \left(\frac{1}{N}\right)^2 \text{cov}(r_i, r_j)$$

Considerando que para  $N$  ativos temos  $N(N-1)$  pares de covariâncias:

$$E[\text{cov}(r_i, r_j)] = \frac{\sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \text{cov}(r_i, r_j)}{N(N-1)} \Leftrightarrow \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \text{cov}(r_i, r_j) = N(N-1)E[\text{cov}(r_i, r_j)]$$

Rescrevendo a variância média:

$$\begin{aligned}\text{var}[R] &= \frac{1}{N} E[\sigma_i^2] + \frac{1}{N^2} N(N-1) E[\text{cov}(r_i, r_j)] = \\ &= \frac{E[\sigma_i^2]}{N} + \frac{(N-1)}{N} E[\text{cov}(r_i, r_j)]\end{aligned}$$

O que dá:

$$\text{var}[R] = \frac{\sum_{i=1}^N \text{var}(r_i)}{N^2} + \frac{(N-1)}{N} \times \text{cov média}$$

Onde:

$$\begin{aligned} \lim_{N \rightarrow \infty} \text{var}[R] &= \lim_{N \rightarrow \infty} \left\{ \frac{E[\sigma_i^2]}{N} + \frac{(N-1)}{N} E[\text{cov}(r_i, r_j)] \right\} = \\ &= \lim_{N \rightarrow \infty} \left\{ \frac{E[\sigma_i^2]}{N} \right\} + \lim_{N \rightarrow \infty} \left\{ \frac{N}{N} E[\text{cov}(r_i, r_j)] \right\} - \lim_{N \rightarrow \infty} \left\{ \frac{1}{N} E[\text{cov}(r_i, r_j)] \right\} = \\ &= E[\text{cov}(r_i, r_j)] \end{aligned}$$

Assim, quando  $N$  aumenta, a primeira parcela da variância tende a zero, porém a segunda permanece.

A existência desse resíduo, que não pode ser eliminado por diversificação, origina a uma classificação para os tipos de risco:

- Risco próprio: atribuído a eventos específicos do ativo. Também denominado risco idiosincrático, específico, não sistemático ou diversificável.
  
- Risco sistemático: risco aos quais todos os ativos estão expostos, portanto a diversificação é inócua para este tipo de risco.

## ANEXO 2 - Algoritmo de Preparação dos Dados

### **% Logaritmo dos Retornos**

```
tic
for i= 1:199
    for j=1:25
        retorno(i,j)=log(data(200-i+1,j)/data(200-i,j));
    end
end
```

### **%Substituindo os "NaNs" e "Inf"**

```
for i= 1:199
    for j=1:25
        if (retorno(i,j)==-Inf)
            retorno(i,j)=-1;
        elseif (isnan(retorno(i,j)));
            retorno(i,j)=0;
        end
    end
end
```

### **%Mantendo a Matriz Positiva Definida**

```
for i= 1:199
    for j=1:25
        if (retorno(i,j)==0)
            retorno(i,j)=0.00001;
        end
    end
end
```

### **%%%%Previsao dos Retornos%%%%**

```
cov1=cov(retorno(1:109,:));
a=chol(cov1)';
b=randn(90,25);
prev(1,:)=b(1,:)*a;
```

### **%Garch da Matriz de Covariancia com lambda=0.94%**

```
for k=1:89
    for i=1:25
        for j=1:25
            cov1(i,j,k+1)=0.94*cov1(i,j)+0.06*retorno(k+110,i)*retorno(k+110,j);
        end
    end
end
```

### **%Calculo dos retornos**

```
for i=1:89
    a=chol(cov1(:, :, i+1))';
```

```

    prev(i+1,:)=b(i+1,:)*a;
end

```

### **%Populações Iniciais**

#### **%Proporções**

```

for i=1:90;
[x,y,z]=frontcon(prev(i,:),cov1(:, :, i),100);
prop(:, :, i)=z;
end

```

### **%Corrigindo os pequenos valores negativos**

```

for i=1:100
for j=1:25
for k=1:90
if (prop(i,j,k)<0)
    prop(i,j,k)=0;
end
end
end
end

```

### **%Arredondamento**

```

invest=1000000*prop;
for i=1:90
unit(i,:)=data(92-i,1:25).*(1-retorno(i,:))*100;
unit(i,11)=unit(i,11)/10;
unit(i,15)=unit(i,15)*10;
unit(i,19)=unit(i,19)/10;
end

```

```

for j=1:90
for i=1:100
invest(i, :, j)=invest(i, :, j)./unit(j, :);
end
end

```

### **%Arredondando para o menor valor inteiro**

```

invest=floor(invest);

```

### **%Retirando os NaNs**

```

for i=1:100
    for j=1:25
        for k=1:90
if (isnan(invest(i,j,k)));
    invest(i,j,k)=0;
end
end
end
end
t=toc

```

## ANEXO 3 – Algoritmo Genético Simples

```
%Algoritmo Genetico Simples%
tic
corret=zeros(1,25);
maximo=1000000;

for v=1:90
%População Inicial
pop1=invest(:,:,v);

%Pesos das cotações
pesos=zeros(25,1);
pesos=pesos+100;
pesos(11,1)=1000;
pesos(15,1)=10;
pesos(19,1)=1000;

best=zeros(3000,1);
k=1;
for k=1:3000

    %Total Investido
    for i=1:25
        for j=1:100
            pesos1(i,j,1)=pesos(i,1)*pop1(j,i,1);
        end
    end

    for i=1:100;
        pop1(i,26,1)=data(1,:)*pesos1(:,i,1);
    end

    %Cálculo da Corretagem
    for i=1:100
        for j=1:25
            corret(i,j,2)=pop1(i,j,1)-corret(1,j);
        end
    end

    for i=1:100
        for j=1:25
            corret(i,j,3)=corret(i,j,2)*data(91-v,j);
        end
    end

    for i=1:100
        for j=1:25
```



```

if corret(i,j,3)<135.05
    corret(i,j,4)=2.70;
elseif corret(i,j,3)<268.62
    corret(i,j,4)=corret(i,j,3)*0.02;
elseif corret(i,j,3)<1514.68
    corret(i,j,4)=corret(i,j,3)*0.015+2.26;
elseif corret(i,j,3)<3029.37
    corret(i,j,4)=corret(i,j,3)*0.01+10.6;
else
    corret(i,j,4)=corret(i,j,3)*0.005+25.21;
end
end
end

for i=1:100
    corret(i,26,4)=sum(corret(i,1:25,4));
end

for i=1:100
    if (isnan(corret(i,26,4)));
        corret(i,26,4)=0;
    end
end

%Retorno Esperado
for i=1:100;
    pop1(i,50,1)=prev(1,:)*pesos1(:,i)-corret(i,25,4);
end

%DesvPad Esperado
for i=1:100;
    for j=1:25
        vetor(j,1)=data(1,j)*pesos1(j,i);
    end
    pop1(i,51,1)=vetor'*cov1(:,,1)*vetor;
end

for i=1:100;
    pop1(i,51,1)=sqrt(pop1(i,51,1));
end

%Calculo do Indice
for i=1:100
    if (pop1(i,51,1)>0)
        pop1(i,52,1)=pop1(i,50,1)/pop1(i,51,1);
    else
        pop1(i,52,1)=0;
    end
end
end

```

```

%Penalização por Estourar o Orçamento
for i=1:100
    if (pop1(i,26,1)>maximo)
        pop1(i,52,1)=pop1(i,52,1)-100000;
    end
end

%Ordenando pelo indice
pop1(:,1)=sortrows(pop1(:,1),52);

%Elitismo
pop1(1,2)=pop1(100,1);
pop1(51,2)=pop1(100,1);

%Gravando a Melhor Solução
best(k,1)=pop1(100,52,1);

%Matriz de Cruzamento
mtz=zeros(26,2);
for i=1:26
    for j=1:2
        mtz(i,j)=rand*50;
        mtz(i,j)=round(mtz(i,j));
    end
end

for i=1:26
    for j=1:2
        if(mtz(i,j)==0) mtz(i,j)=1;
        end
    end
end

%Cruzamento
for i=1:26
    pop1(i+1,1:24,2)=pop1(mtz(i,1)+50,1:24,1);
    pop1(i+1,25:25,2)=pop1(mtz(i,2)+50,25:25,1);
    pop1(i+51,1:24,2)=pop1(mtz(i,2)+50,1:24,1);
    pop1(i+51,25:25,2)=pop1(mtz(i,1)+50,25:25,1);
end

%Mutação
for i=2:100
    for j=1:25
        dado=rand;
        if (dado<0.05)
            pop1(i,j,2)=pop1(i,j,2)+round(rand*10-5);
        end
    end
end

```

```
    end
end

% Volta para o primeiro plano
pop1(:, :, 1) = pop1(:, :, 2);
k = k + 1;

end

%% Avaliando Resultado
retreal = retorno(109 + v, :) * pop1(100, 1:25, 1);

% Dinheiro Disponível
maximo = maximo + retreal;
retornofinal(v, 1) = retreal;

v = v + 1;
end

t = toc
```

## ANEXO 4 - MOGA

```
%Multi Objective Genetic Algorithm%
```

```
tic
```

```
corret=zeros(1,50);
```

```
maximo=1000000;
```

```
for v=1:45
```

```
%Populacao Inicial
```

```
pop1=invest(:,v);
```

```
%Pesos das cotações
```

```
pesos=zeros(50,1);
```

```
pesos=pesos+100;
```

```
pesos(42,1)=10;
```

```
pesos(11,1)=1000;
```

```
pesos(15,1)=10;
```

```
pesos(30,1)=10;
```

```
pesos(47,1)=10;
```

```
pesos(46,1)=1000;
```

```
pesos(19,1)=1000;
```

```
best=zeros(3000,1);
```

```
k=1;
```

```
for k=1:3000
```

```
%Total Investido
```

```
for i=1:50
```

```
for j=1:100
```

```
pesos1(i,j,1)=pesos(i,1)*pop1(j,i,1);
```

```
end
```

```
end
```

```
for i=1:100;
```

```
pop1(i,49,1)=data(1,:)*pesos1(:,i,1);
```

```
end
```

```
%Cálculo da Corretagem
```

```
for i=1:100
```

```
for j=1:50
```

```
corret(i,j,2)=pop1(i,j,1)-corret(1,j);
```

```
end
```

```
end
```

```
for i=1:100
```

```
for i=1:50
```

```

    corret(i,j,3)=corret(i,j,2)*data(91-v,j);
end
end

for i=1:100
for i=1:50
if corret(i,j,3)<135.05
    corret(i,j,4)=2.70;
elseif corret(i,j,3)<498.62
    corret(i,j,4)=corret(i,j,3)*0.02;
elseif corret(i,j,3)<1514.68
    corret(i,j,4)=corret(i,j,3)*0.015+2.49;
elseif corret(i,j,3)<3029.37
    corret(i,j,4)=corret(i,j,3)*0.01+10.6;
else
    corret(i,j,4)=corret(i,j,3)*0.005+25.21;
end
end
end

for i=1:100
    corret(i,49,4)=sum(corret(i,1:50,4));
end

for i=1:100
if (isnan(corret(i,49,4)));
    corret(i,49,4)=0;
end
end

%Retorno Esperado
for i=1:100;
    pop1(i,50,1)=prev(1,:)*pesos1(:,i)-corret(i,49,4);
end

%DesvPad Esperado

for i=1:100;
    for j=1:50
        vetor(j,1)=data(1,j)*pesos1(j,i);
    end
    pop1(i,51,1)=vetor'*cov1(:,,1)*vetor;
end

for i=1:100;
    pop1(i,51,1)=sqrt(pop1(i,51,1));
end

for i=1:100;
if(pop1(i,51,1)==0)

```

```

    pop1(i,51,1)=0.01;
end
end

```

#### **%Penalização por Estourar o Orçamento**

```

for i=1:100
    if (pop1(i,49,1)>1000000)
        pop1(i,50,1)=pop1(i,50,1)-100000;
    end
end

```

#### **%Calculando o Rank das Soluções%**

```

for i=1:100
    pop1(i,52,1)=1;
end

```

#### **%Numero de solucoes que dominam%**

```

for i=1:100
    for j=1:100
        if pop1(i,51,1)==pop1(j,51,1) && pop1(i,50,1)<pop1(j,50,1);
            pop1(i,52,1)=pop1(i,52,1)+1;
        elseif pop1(i,51,1)> pop1(j,51,1) && pop1(i,50,1)<pop1(j,50,1);
            pop1(i,52,1)=pop1(i,52,1)+1;
        elseif pop1(i,51,1)> pop1(j,51,1) && pop1(i,50,1)==pop1(j,50,1);
            pop1(i,52,1)=pop1(i,52,1)+1;
        end
    end
end
end

```

#### **%Raw Fitness**

```

for i=1:100
    pop1(i,53,1)=pop1(i,50,1)/pop1(i,51,1);
end

```

#### **%Elitismo**

```

pop1(:, :, 1)=sortrows(pop1(:, :, 1), 53);
pop1(1, :, 2)=pop1(100, :, 1);
pop1(51, :, 2)=pop1(100, :, 1);

```

#### **%Contadores dos Rankings**

```

for i=1:100
    rank(i,1)=i;
end

```

#### **%%% Distancia e Fitness Sharing%**

```

for i=1:100
    nElements=length(pop1(:,52,1));
    IndexIDs=1:nElements;
    Result=eval(['pop1(:,52,1)==rank(i,1)']);

```

```

Result=IndexIDs(Result);
rank(i,2)=length(Result);
rank(i,3)=sum(pop1(Result,53,1));
if (rank(i,2)==0)
    rank(i,4)=0;
else
rank(i,4)=rank(i,3)/rank(i,2);
end

```

### **%Distancia entre as soluções**

```

share=pop1(Result,50:51,1);
sssret=max(share(:,1))-min(share(:,1));
sssdp=max(share(:,2))-min(share(:,2));
for i=1:length(Result);
    for j=1:length(Result);
        if (sssret==0)
            mdist(i,j)=sqrt((((share(i,1)-share(j,1)))^2)+(((share(i,2)-share(j,2))))^2));
        elseif (sssdp==0)
            mdist(i,j)=sqrt((((share(i,1)-share(j,1)))^2)+(((share(i,2)-share(j,2))))^2));
        else
            mdist(i,j)=sqrt((((share(i,1)-share(j,1))/sssret)^2)+(((share(i,2)-share(j,2))/sssdp)^2));
        end
    end
end
end

mdist2=nansum(mdist);

```

### **%Aptidao Compartilhada**

```

for i=1:length(Result)
pop1(Result(1,i),54,1)=rank(1,4)*mdist2(1,i);
end

```

end

```

pop1(:,1)=sortrows(pop1(:,1),53);

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

### **%Operações normais de um Algoritmo Genetico Simples%**

```

best(k,1)=pop1(100,53,1);

```

### **%Matriz de Cruzamento**

```

mtz=zeros(49,2);
for i=1:49
    for j=1:2
        mtz(i,j)=rand*50;
        mtz(i,j)=round(mtz(i,j));
    end
end

```

```

end

for i=1:49
    for j=1:2
        if(mtz(i,j)==0) mtz(i,j)=1;
        end
    end
end

%Cruzamento
for i=1:49
    pop1(i+1,1:24,2)=pop1(mtz(i,1)+50,1:24,1);
    pop1(i+1,25:50,2)=pop1(mtz(i,2)+50,25:50,1);
    pop1(i+51,1:24,2)=pop1(mtz(i,2)+50,1:24,1);
    pop1(i+51,25:50,2)=pop1(mtz(i,1)+50,25:50,1);
end

%Mutaçao
for i=2:100
    for j=1:50
        dado=rand;
        if (dado<0.25)
            pop1(i,j,2)=pop1(i,j,2)+round(rand*10-5);
        end
    end
end

%Volta para o primeiro plano
pop1(:,1)=pop1(:,2);

%% Avaliando Resultado
retreal=retorno(109+v,:)*pop1(100,1:50,1)';

%Dinheiro Disponível

maximo=maximo+retreal;

retornofinal(v,1)=retreal;
k=k+1;

end
end

t=toc;
plot(best)

```



## ANEXO 5 NSGA

```
%Non Dominated Sorting Genetic Algorithm %
```

```
corret=zeros(1,50);  
maximo=1000000;
```

```
for v=1:90
```

```
%População Inicial
```

```
pop1=invest(:,:,v);
```

```
%Pesos das cotações
```

```
pesos=zeros(50,1);  
pesos=pesos+100;  
pesos(42,1)=10;  
pesos(11,1)=1000;  
pesos(15,1)=10;  
pesos(30,1)=10;  
pesos(47,1)=10;  
pesos(46,1)=1000;  
pesos(19,1)=1000;
```

```
for k=1:3000
```

```
%Total Investido
```

```
for i=1:50
```

```
for j=1:100
```

```
pesos1(i,j,1)=pesos(i,1)*pop1(j,i,1);
```

```
end
```

```
end
```

```
for i=1:100;
```

```
pop1(i,49,1)=data(1,:)*pesos1(:,i,1);
```

```
end
```

```
%Cálculo da Corretagem
```

```
for i=1:100
```

```
for j=1:50
```

```
corret(i,j,2)=pop1(i,j,1)-corret(1,j);
```

```
end
```

```
end
```

```
for i=1:100
```

```
for i=1:50
```

```
corret(i,j,3)=corret(i,j,2)*data(91-v,j);
```

```
end
```

```
end
```

```

for i=1:100
for i=1:50
if corret(i,j,3)<135.05
    corret(i,j,4)=2.70;
elseif corret(i,j,3)<498.62
    corret(i,j,4)=corret(i,j,3)*0.02;
elseif corret(i,j,3)<1514.68
    corret(i,j,4)=corret(i,j,3)*0.015+2.49;
elseif corret(i,j,3)<3029.37
    corret(i,j,4)=corret(i,j,3)*0.01+10.6;
else
    corret(i,j,4)=corret(i,j,3)*0.005+25.21;
end
end
end

for i=1:100
    corret(i,49,4)=sum(corret(i,1:50,4));
end

for i=1:100
if (isnan(corret(i,49,4)));
    corret(i,49,4)=0;
end
end

%Retorno Esperado
for i=1:100;
    pop1(i,50,1)=prev(1,:)*pesos1(:,i)-corret(i,49,4);
end

%DesvPad Esperado

for i=1:100;
    for j=1:50
        vetor(j,1)=data(1,j)*pesos1(j,i);
    end
    pop1(i,51,1)=vetor'*cov1(:,,1)*vetor;
end

for i=1:100;
    pop1(i,51,1)=sqrt(pop1(i,51,1));
end

%Penalizaçao por Estourar o Orçamento

for i=1:100
    if (pop1(i,49,1)>1000000)
        pop1(i,50,1)=pop1(i,50,1)-100000;
    end
end

```

```

end

for i=1:100
    if (pop1(i,51,1)==0)
        pop1(i,51,1)=0.001;
    end
end

for i=1:100
    pop1(i,53,1)=pop1(i,50,1)/pop1(i,51,1);
end

%Calculando o Rank das Soluções%

for i=1:100
    pop1(i,52,1)=1;
end

%Numero de soluções que dominam%

for i=1:100
    for j=1:100
        if pop1(i,51,1)==pop1(j,51,1) && pop1(i,50,1)<pop1(j,50,1);
            pop1(i,52,1)=pop1(i,52,1)+1;
        elseif pop1(i,51,1)> pop1(j,51,1) && pop1(i,50,1)<pop1(j,50,1);
            pop1(i,52,1)=pop1(i,52,1)+1;
        elseif pop1(i,51,1)> pop1(j,51,1) && pop1(i,50,1)==pop1(j,50,1);
            pop1(i,52,1)=pop1(i,52,1)+1;
        end
    end
end

pop1(:, :, 1)=sortrows(pop1(:, :, 1), 52);

%Nova população
for i=1:50
    pop1(i, :, 2)=pop1(i, :, 1);
end

pop1(:, :, 1)=sortrows(pop1(:, :, 1), 53);
pop1(1, :, 2)=pop1(100, :, 1);
pop1(51, :, 2)=pop1(100, :, 1);

%%%%%%%%% Calculo da Crowding Distance%

front=pop1(50, 52, k+2);

nElements=length(pop1(:, 52, k));

```

```

IndexIDs=1:nElements;
Result=eval(['pop1(:,52,k)==front']);
Result=IndexIDs(Result);
Result1=eval(['pop1(:,52,k+1)==front']);
Result1=IndexIDs(Result1);

%Crowding Distance das Soluções%%

```

```

for i=1:length(Result)
    share(i,1)=Result(1,i);
end
share(:,2:3)=pop1(Result,50:51,k);
sssret=max(share(:,2))-min(share(:,2));
sssdp=max(share(:,3))-min(share(:,3));
sortrows(share,3);
share(1,4)=Inf;
share(length(share),4)=Inf;

for i=1:length(share);
    for j=1:length(Result);
        mdist(i,j)=((share(i,3)-share(j,3))/sssdp);
    end
end
mdist2=nansum(mdist);
share(:,4)=share(:,4)+mdist2';

share=sortrows(share,4);

```

**%Substituição dos indivíduos da ultima fronteira escolhida**

```

for i=1:length(Result1)
    pop1(51-i,:,k+1)=pop1(share(length(share),1)-i,:,k);
end

end

```

**%Cálculos utilizados do Algoritmo Genetico Simples%**

```

best(k,1)=pop1(100,53,1);

```

**%Matriz de Cruzamento**

```

mtz=zeros(49,2);
for i=1:49
    for j=1:2
        mtz(i,j)=rand*50;
        mtz(i,j)=round(mtz(i,j));
    end
end
end

```

```

for i=1:49
    for j=1:2
        if(mtz(i,j)==0) mtz(i,j)=1;
        end
    end
end

%Cruzamento
for i=1:49
    pop1(i+1,1:24,2)=pop1(mtz(i,1)+50,1:24,1);
    pop1(i+1,25:50,2)=pop1(mtz(i,2)+50,25:50,1);
    pop1(i+51,1:24,2)=pop1(mtz(i,2)+50,1:24,1);
    pop1(i+51,25:50,2)=pop1(mtz(i,1)+50,25:50,1);
end

%Mutacao
for i=2:100
    for j=1:50
        dado=rand;
        if (dado<0.2)
            pop1(i,j,2)=pop1(i,j,2)+round(rand*10-5);
        end
    end
end

%Volta para o primeiro plano

pop1(:,1)=pop1(:,2);

%% Avaliando Resultado
retreal=retorno(109+v,:)*pop1(100,1:50,1);

%Dinheiro Disponível

maximo=maximo+retreal;

retornofinal(v,1)=retreal;

k=k+1;

end
v=v+1
end
plot(best)

```

# 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](#)