

INDEX TRACKING COM CONTROLE DO NÚMERO DE ATIVOS E APLICAÇÃO COM ALGORITMOS GENÉTICOS

Leonardo Riegel Sant'Anna

Universidade Federal do Rio Grande do Sul
lrsantanna@outlook.com

Tiago Pascoal Filomena

Universidade Federal do Rio Grande do Sul
tpfilomena@ea.ufrgs.br

Pablo Cristini Guedes

Universidade Federal do Rio Grande do Sul
pabloguedes@gmail.com

Denis Borenstein

Universidade Federal do Rio Grande do Sul
dborenstein@ea.ufrgs.br

RESUMO

Neste artigo, discutimos a estratégia de investimento de *Index Tracking*. Nosso objetivo é solucionar um modelo de otimização de *index tracking* aplicando Algoritmos Genéticos (AG) para formação de carteiras de número reduzido de ativos com baixo tempo computacional. O índice adotado é o Ibovespa, entre Jan/2009 a Jul/2012. Os intervalos de rebalanceamento são mensal, trimestral, semestral e anual. Estudos prévios permitiram verificar que, com otimização sem uso de heurísticas, o modelo proposto dificilmente pode ser solucionado para formação de carteiras de 20 ativos com tempo máximo de processamento de uma hora e *gap* médio das soluções abaixo de 10%. Neste artigo, usando Algoritmos Genéticos, formamos carteiras de 5 e 10 ativos com tempo de processamento em torno de 5 minutos, obtendo soluções com *gap* médio estimado abaixo de 10%. E, com tempo um pouco maior, em torno de 8 minutos, obtivemos soluções com *gap* médio abaixo de 5%.

PALAVRAS CHAVE: index track, otimização de carteiras, algoritmos genéticos.

Área principal: Gestão Financeira.

ABSTRACT

In this paper, we discuss the implementation of an index tracking strategy. Our goal is to solve index tracking problems with limited number of assets using Genetic Algorithms (AG). The benchmark is the IBOVESPA index between jan/2009 and jul/2012. We chose four different windows to rebalance the portfolio: monthly, quarterly, semiannual and annual. Previous works have shown that, without using a heuristic approach, those problems could not be solved to form portfolios with a constraint of 20 assets and one hour of computing time with average GAP below 10%. In this study, we provide portfolios with 5 and 10 assets with computational time under five minutes and average MIP GAP below 10%. Relaxing the computing time for 8 minutes, the average MIP Gap was on average below 5%.

KEYWORDS: index track, portfolio optimization, genetic algorithm.

1. Introdução

A Teoria Moderna de Portfólio iniciou-se com Markowitz (1952), que propôs um modelo de média-variância para seleção de carteiras de investimento. Pela literatura, podemos verificar que há basicamente dois tipos de fundos de investimentos: fundos de administração ativa e de administração passiva (Roll, 1992; Beasley, Meade e Chang, 2003; Maringer e Oyewumi, 2007).

Fundos ativos têm maior quantidade de operações de compra e venda de ativos e, por consequência, maiores custos de transação. Já fundos passivos baseiam-se na hipótese de eficiência dos mercados proposta por Fama (1970); segundo esta hipótese, o rendimento do mercado não tende a ser superado por fundos ativos (Frino e Gallagher, 2001; Fama e French, 2010). Fundos passivos têm gestão menos flexível, pois apresentam menor quantidade de transações de compra e venda e, portanto, menores custos de transação e gestão.

Fundos de *index tracking* (*index funds*, ou “fundos de índice” no Brasil) são um método de fundo passivo que busca reproduzir o rendimento de um índice de mercado (por exemplo: índice Ibovespa). Essa estratégia pode ser usada, por exemplo, para formação de um ETF (do inglês, *Exchange-Traded Fund* – ativo que representa um fundo de índice, e que tem se popularizado no Brasil nos últimos anos – Borges, Eid Jr. e Yoshinaga, 2012) ou para reprodução de um indicador do mercado, como inflação (em vez de seguir um índice).

A primeira opção para formação de um *index fund* seria realizar uma réplica exata do índice. Porém, essa estratégia tem a desvantagem de formar carteiras de grande quantidade de ativos, tendo assim maior quantidade de transações e maiores custos (Barro e Canestrelli, 2009; Canakgoz e Beasley, 2009). Assim, os modelos de *index tracking* comumente adotam a limitação de quantidade de ativos na carteira, para reduzir custos de transação e de gestão. Neste estudo, adotamos um modelo de *index tracking* derivado de Gaivoronski, Krylov e van der Wijst (2005), com restrição do número de ativos. Utilizamos estes autores como referência por ser um estudo amplamente referenciado, apresentando diferentes abordagens para o problema de *index tracking*.

Assim, nosso foco é aplicar um modelo com objetivo de minimizar a diferença entre rendimento dos portfólios e do índice (minimização do *tracking error*), com restrição da quantidade de ativos nas carteiras. Para isso, o erro de *tracking* de cada carteira corresponde à diferença entre o rendimento acumulado da carteira e o rendimento acumulado do índice em cada intervalo de tempo fora da amostra.

Neste artigo, adotamos o índice Ibovespa como referência, para o período de Jan/2009 a Jul/2012, aplicando essa abordagem de *index tracking* ao ambiente brasileiro. Com uma amostra de 67 ativos, nosso objetivo é formar carteiras de 5 e 10 ativos com tempo computacional inferior a 10 minutos e *gap* máximo das soluções de 10% (em linha com os tempos de solução de Beasley, Meade e Chang, 2003). Assim, este estudo trata-se de uma extensão de Sant’Anna, Filomena e Borenstein (2013), em que se verificou a impossibilidade de formar carteiras de 20 ativos (para essa amostra de dados) com tempo de otimização inferior a 1 hora e *gap* abaixo de 10%. Os períodos de rebalanceamento das carteiras são 20, 60, 120 e 240 dias úteis (basicamente, rebalanceamento mensal, trimestral, semestral e anual).

Neste estudo, usamos o método de algoritmos genéticos para formação de carteiras de número reduzido de ativos com baixo tempo de processamento. A necessidade de se formar portfólios menores justifica-se pelo entendimento de que, em situações reais de uma empresa, decisões sobre a gestão da carteira precisam ser tomadas com agilidade. Situações como a alteração da carteira virtual do índice, por exemplo, podem exigir rapidez na análise para atualização do portfólio de *index tracking*. Além disso, em períodos de maior volatilidade do mercado, pode ser preciso realizar diversos testes para a tomada de decisão sobre o rebalanceamento da carteira, para que se verifique, por exemplo, se é melhor que a nova carteira de *tracking* tenha uma quantidade pequena de ativos, ou uma quantidade um pouco maior, com base nas previsões de erro de *tracking* futuro. No caso do mercado brasileiro, que possui maior

volatilidade que mercados como o americano, torna-se importante essa capacidade de resolver a otimização com maior velocidade, principalmente em momentos de maior incerteza no mercado.

A escolha de algoritmos genéticos baseia-se nas boas soluções obtidas por Beasley, Meade e Chang (2003), Oh, Kim e Min (2005) e Jeurissen e van den Berg (2008). Como diferencial, analisamos a qualidade das soluções através da verificação dos *gaps* das respostas do algoritmo, o que não é feito de forma recorrente na literatura. Pelos resultados na seção 4, constatamos a qualidade da heurística, com soluções de *gap* médio inferior a 10% para tempo de processamento em torno de 5 minutos, e *gap* médio inferior a 5% para tempo em torno de 8 minutos. Pelos *gaps*, notamos a tendência de se obter, respostas ao menos próximas das respostas ótimas.

Este artigo desenvolve-se da seguinte forma: na seção 2, apresentamos uma breve análise da literatura contemporânea sobre *index tracking*; na seção 3, descrevemos a metodologia; na seção 4, demonstramos os principais resultados; e, na seção 5, tratamos das conclusões do estudo.

2. Revisão Literária

Na primeira subseção da revisão literária, apresentamos alguns exemplos de trabalhos de *index tracking* com métodos de solução variados. E, na segunda subseção, apresentamos alguns exemplos de estudos de *index tracking* com uso de algoritmos genéticos.

2.1. Otimização de Portfolios de Investimento de *Index Track*

O problema de *index tracking* é tratado, na literatura contemporânea, com diversos métodos de solução como, por exemplo, uso de métodos heurísticos (Beasley, Meade e Chang, 2003; Oh, Kim e Min, 2005; Maringer e Oyewumi, 2007; Jeurissen e van den Berg, 2008; Guastaroba e Speranza, 2012; Scozzari et al, 2012), uso de cointegração (Dunis e Ho, 2005; Caldeira e Portugal, 2010) e uso de programação quadrática (Jansen e van Dijk, 2002; Coleman, Li e Henniger, 2006).

Gaivoronski, Krylov e van der Wijst (2005) discutem diferentes abordagens de *index tracking*, com restrições para análise de risco do portfólio, rebalanceamento das carteiras e restrição de custos de transação. Estes autores verificam que o erro de *tracking* tende a ser menor com uso de maiores intervalos de dados dentro da amostra, além de que o desempenho das carteiras é influenciado pela quantidade de ativos da carteira e pela forma de realização de rebalanceamentos.

Dunis e Ho (2005) e Caldeira e Portugal (2010) utilizam cointegração para *index tracking*, pois esse método incorpora “comovimentos” de longo prazo nos preços dos ativos, tendo assim a tendência de gerar carteiras mais estáveis, o que implica menos rebalanceamentos e menores custos. Os melhores resultados foram obtidos para rebalanceamento trimestral.

Jansen e van Dijk (2002) e Coleman, Li e Henniger (2006) fazem uso de programação quadrática para *index tracking* com reduzido número de ativos, porém sem uso de restrição inteira para controlar o tamanho das carteiras; sendo assim, a definição dos ativos de cada carteira é feita de forma exógena, antes da realização da otimização do modelo. Conforme os autores, esse problema possui complexidade computacional NP-Hard, e o tempo de processamento é elevado para formação de carteiras reduzidas.

Maringer e Oyewumi (2007) e Scozzari et al (2012) aplicam modelagens de programação quadrática e inteira, com controle da quantidade de ativos. Para solução, ambos usam heurísticas de evolução diferencial, principalmente devido ao uso de restrição inteira. Os testes permitiram verificar a tendência de se obter menores erros de *tracking* com carteiras maiores. Da mesma forma, Guastaroba e Speranza (2012) usam formulação linear e inteira e uma heurística de evolução diferencial chamada Kernel Search. Nos testes, foram utilizados oito índices com amostras de até 2.151 ativos e formação de carteiras com no máximo 90 ativos;

assim, nota-se novamente a necessidade de um método heurístico para formação de carteiras reduzidas proporcionalmente ao tamanho da amostra.

2.2. Uso de Algoritmos Genéticos para *Index Track*

Pela revisão realizada, foi possível notar a relevância de heurísticas evolucionárias e algoritmos genéticos (AG) para *index tracking*, pois são métodos recorrentes na literatura. Maringer e Oyewumi (2007) e Scozzari et al (2012) são exemplos de uso do método de evolução diferencial, e outros autores que usam AG são Beasley, Meade e Chang (2003), Oh, Kim e Min (2005) e Jeurissen e van den Berg (2008).

Beasley, Meade e Chang (2003) são uma das referências mais citadas sobre o assunto, e apresentam uma formulação de *index tracking* com restrição de quantidade de ativos, restrição de custos de transação e controle de rebalanceamento; para solução, desenvolvem um método de “heurística populacional”, considerando a complexidade do problema e a necessidade de se obter respostas mais rápidas. Foram realizados testes com os índices Hang Seng (amostra de 31 ativos), DAX (85 ativos), FTSE (89 ativos), S&P 100 (98 ativos) e Nikkei (225 ativos). O tempo médio ficou em 6,7 minutos (sendo que o menor tempo foi 1,7 minutos e o maior foram 15,7 minutos).

Oh, Kim e Min (2005) desenvolveram um AG para *index tracking* separando o algoritmo em duas partes. O primeiro passo do algoritmo trata de selecionar os ativos que compõem o portfólio, com base em indicadores das empresas. Após essa seleção, é realizada a minimização de uma função objetivo utilizando como amostra apenas os ativos previamente selecionados. A eficiência do AG é demonstrada via análise dos erros de *tracking*. Os tempos de solução também são analisados, usando em média em torno de 10 minutos de processamento. A rapidez na obtenção das soluções, combinada com os baixos valores de erro, demonstra a eficiência do método.

Outro exemplo de uso de AG é Jeurissen e van den Berg (2008). Esses autores utilizam os softwares Matlab (para codificação do algoritmo, com uso de função quadrática) e Microsoft Excel. Os testes envolveram formar carteiras de 10 ativos com uma amostra de 25 ativos. Uma das conclusões é que, dependendo dos parâmetros adotados para o AG, o algoritmo tende a encontrar a solução ótima com até 23 gerações no caso da amostra utilizada.

3. Metodologia

Para melhor demonstração da metodologia empregada, apresentamos na subseção 3.1 o modelo de otimização utilizado; na seção 3.2, é explicada a base de dados; e, na seção 3.3, é feita a descrição do método de aplicação da heurística de algoritmos genéticos. Para desenvolvimento do modelo, utilizamos a seguinte notação:

x_i = variável de peso do ativo i na carteira, sendo $0 \leq x_i \leq 1$;

z_i = variável binária para cada ativo i ;

r_{it} = parâmetro: retorno do ativo i no momento de tempo t ;

R_t = parâmetro: retorno do índice no momento de tempo t ;

t = momento de tempo (cada dia útil com negociação de ativos no mercado);

T = total de momentos de tempo;

N = quantidade de ativos da amostra;

K = quantidade limite de ativos na carteira;

ϑ = parâmetro: limite mínimo de erro entre carteira e índice em cada t ;

θ = parâmetro: limite máximo de erro entre carteira e índice em cada t ;

3.1. Modelo de Otimização

O modelo utilizado foi elaborado com base na função objetivo descrita em Gaivoronski, Krylov e van der Wijst (2005). Nesse modelo, temos duas variáveis de decisão, x_i e z_i . A

variável x_i representa o peso de cada ativo i na carteira (ou seja, x representa a composição do portfólio); e a variável z_i é binária, com valor 1 se o ativo i é incluído na carteira, e valor 0 caso contrário. A função objetivo busca minimizar a variância da diferença entre retorno do portfólio e do índice.

$$\text{Minimizar } \frac{1}{T} \sum_{t=1}^T \left[\left(\sum_{i=1}^I x_i \cdot r_{it} \right) - R_t \right]^2 \quad (1)$$

$$\text{Sujeito a } \sum_{i=1}^I (x_i \cdot r_{it}) - R_t \geq \vartheta, \forall t \quad (2)$$

$$\sum_{i=1}^I (x_i \cdot r_{it}) - R_t \leq \theta, \forall t \quad (3)$$

$$\sum_{i=1}^I x_i = 1 \quad (4)$$

$$\sum_{i=1}^I z_i \leq K \quad (5)$$

$$x_i \leq z_i, \forall i \quad (6)$$

$$x_i \geq 0, \forall i \quad (7)$$

$$z_i = 0; 1, \forall i \quad (8)$$

As restrições (2) e (3) limitam a diferença de rendimento entre carteira e índice a um valor mínimo e a um valor máximo para cada momento t (parâmetros ϑ e θ). A restrição (4) define que 100% do valor disponível deve ser utilizado. As restrições (5) e (6), em conjunto, definem o número máximo de ativos da carteira (parâmetro K). As restrições (7) e (8) definem, respectivamente, que o valor de x_i deve ser entre 0 e 1 e que z_i é variável binária.

A base de dados é formada pelos retornos diários do índice e dos ativos que compõem o índice, conforme citado na seção seguinte. O retorno diário do índice Ibovespa, para cada dia t (R_t), é calculado com o valor médio do índice nos dias t e $t - 1$ (dia atual e dia anterior). E o retorno diário de cada ativo, em cada t (r_{it}), é calculado com o preço médio do ativo em t e $t - 1$.

$$R_t = \frac{\text{índice_médio}_t}{\text{índice_médio}_{t-1}} - 1 \quad r_{it} = \frac{\text{preço_médio}_{i,t}}{\text{preço_médio}_{i,t-1}} - 1$$

3.2. Base de Dados e Descrição do Problema

A amostra é formada pelos 67 ativos componentes do índice Ibovespa no período de maio/2012 a agosto/2012 – sendo que o índice possui 69 ativos; porém, dois foram retirados por falta de dados (ativos excluídos: ALLL3 e OIBR3).

Nos testes, adotamos $T = 150$ (período dentro da amostra, ou amostra *in-sample*), o que corresponde a um intervalo de sete a oito meses; assim, seguimos a recomendação de Gaivoronski, Krylov e van der Wijst (2005), sobre uso de amostras *in-sample* mais longas para formar carteiras mais estáveis. Cada carteira formada é projetada sobre os retornos de n períodos de tempo subsequentes; nessas projeções, adotamos n igual a 20, 60, 120 e 240 (rebalanceamento mensal, trimestral, semestral e anual). Assim, para formar a primeira carteira para $n = 60$, por exemplo, usamos os dados no intervalo $1 \leq t \leq 150$, e o portfólio é projetado sobre o período fora da amostra em $150 \leq t \leq 210$. A segunda carteira será formada com dados de $61 \leq t \leq 210$, e o portfólio é projetado no período fora da amostra em $210 \leq t \leq 270$, e assim sucessivamente. São formadas, no total, 36 carteiras para $n = 20$, 12 para $n = 60$, 6 para $n = 120$ e 3 para $n = 240$.

3.3. Adaptação do Método de Algoritmos Genéticos ao Problema de *Index Track*

Na aplicação do AG, temos uma população de indivíduos (ou cromossomos) que são possíveis soluções do modelo. Um valor de aptidão (ou *fitness*) é atribuído à cada indivíduo. Para solução, são executadas diversas fases de evolução (gerações), em que os indivíduos sofrem cruzamentos e mutações para geração de novos indivíduos, e os indivíduos de maior aptidão são mantidos na população e carregados para a fase de evolução seguinte – aplicando-se, assim, o conceito de elitismo. Na adaptação do AG, usamos as seguintes definições:

- População (*Num_Popul*): conjunto de P indivíduos;
- Indivíduo: vetor binário de 67 posições que representam a variável $z_i, \forall i = 1, 2, \dots, 67$. Para formação de carteiras de 5 ativos, teremos cinco posições iguais a 1; e, para carteiras de 10 ativos, teremos dez posições iguais a 1;
- Fase de evolução (*Num_Evol*): cada ciclo em que os P indivíduos sofrem cruzamentos e mutações para formação de novos indivíduos;
- Cruzamento (ou *crossover*): dois indivíduos são alterados para formação de dois novos indivíduos, conforme figura 1 abaixo (trata-se da mesma forma de cruzamento usada, por exemplo, por Jeurissen e van den Berg, 2008). Para isso, uma posição de corte é definida aleatoriamente. Assim, a partir de dois indivíduos 1 e 2, são formados um indivíduo 3, com um pedaço inicial do indivíduo 1 e um pedaço final do indivíduo 2 (linha tracejada), e um indivíduo 4, com parte inicial do indivíduo 2 e parte final do indivíduo 1;

		Posição de "Corte"														
		z_1	z_2	z_3	z_4	z_5	z_6	z_7	z_8	z_9	z_{10}	...	z_{65}	z_{66}	z_{67}	
Indivíduo 1		1	1	1	0	0	0	0	0	0	0	...	1	1	0	
Indivíduo 2		0	0	0	1	1	0	0	0	0	1	...	0	0	1	
Indivíduo 3		1	1	1	0	0	0	0	0	0	1	...	0	0	1	
Indivíduo 4		0	0	0	1	1	0	0	0	0	0	...	1	1	0	

Figura 1: representação de um cruzamento. Fonte: elaborado pelos autores.

- Mutação (*Num_Mut*): processo em que cada indivíduo tem uma posição alterada de 0 para 1, e uma posição alterada de 1 para 0. As posições de alteração são definidas de forma aleatória. A mutação pode ainda ser “dupla”, ou seja, alteração de quatro genes, e não de apenas dois. O parâmetro *Num_Mut* será igual a 1 para mutação “simples” (alteração de duas posições), e 2 para mutação “dupla” (alteração de quatro posições);
- Taxa de ocorrência de cruzamentos (*Taxa_Crossover*): probabilidade de que cada dois indivíduos selecionados sofram cruzamento entre si. Adotamos essa taxa como 100% em todos os casos, garantindo a geração de uma quantidade maior de novos indivíduos;
- Taxa de ocorrência de mutações (*Taxa_Mut*): probabilidade de que os indivíduos sofram mutação;
- Aptidão (*fitness*) de cada indivíduo: corresponde ao resultado da função objetivo desse indivíduo. Como estamos utilizando um problema de minimização, os melhores valores de aptidão correspondem aos valores mais próximos de zero. Quanto menor for o valor de aptidão, melhor é o indivíduo. Os valores de aptidão são obtidos conforme explicado a seguir no fluxo de execução do algoritmo.

A metaheurística foi aplicada através da execução de diversas fases de evolução (gerações). Cada geração é formada conforme descrito abaixo e demonstrado na figura 2.

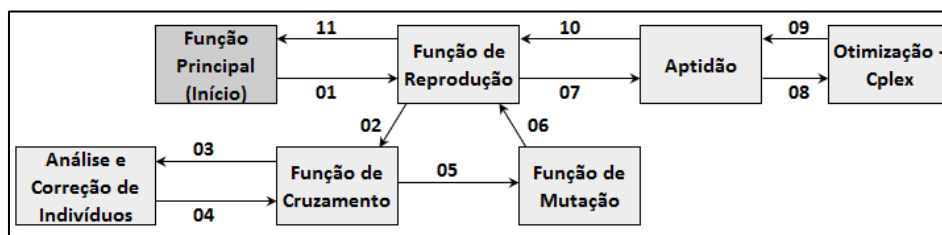


Figura 2: fluxo resumido de cada fase de evolução. Fonte: elaborado pelos autores.

- 1º. Função Principal (Início): recebe uma população inicial de P indivíduos e envia esta população para a Função de Reprodução;
- 2º. Função de Reprodução: recebe a população de P indivíduos e forma todas as possíveis combinações dois a dois desses indivíduos. Cada par de indivíduos é enviado para Função de Cruzamento para sofrer *crossover*. Dois novos indivíduos são gerados e enviados para Função de Análise e Correção para conferência de que possuem a quantidade correta de ativos (5 ou 10 ativos). Caso a quantidade esteja incorreta, o ajuste é feito alterando-se posições aleatoriamente (por exemplo, na formação de carteiras de 5 ativos: se o vetor tiver seis posições iguais a 1, uma dessas posições será selecionada aleatoriamente e alterada para 0, para que a carteira fique com 5 ativos);
- 3º. Função de Mutação: recebe cada indivíduo formado por *crossover* e o submete à mutação (com uma determinada taxa de probabilidade). Assim, após execução de todos os cruzamentos e mutações, a partir de uma população inicial de P indivíduos, tem-se um total de $2C_2^P + P$ indivíduos (indivíduos iniciais mais os novos indivíduos gerados);
- 4º. Função de Aptidão: recebe os $2C_2^P + P$ indivíduos da Função de Reprodução e realiza otimização de cada um desses indivíduos, para definição dos pesos dos ativos em cada carteira (definição de x_i). Para isso, em cada otimização, o Cplex recebe os valores de z_i (vetor binário que representa cada indivíduo) como parâmetros, e não como variáveis – assim, a restrição inteira é eliminada, e o Cplex determinará apenas o peso de cada ativo (x_i) em cada carteira. Dessa forma, para cada indivíduo, tem-se um resultado da função objetivo, correspondente ao seu valor de aptidão;
- 5º. Os indivíduos são retornados para a Função de Reprodução e ordenados para formar uma nova população com os P melhores indivíduos. Assim, inicia-se a formação de uma nova geração.

4. Resultados

Para realização dos testes, foi utilizado um computador Intel® Core™ i7-3770 @ 3.40GHz e 8GB RAM, linguagem de programação C++ e *solver* IBM ILOG Cplex®. O Cplex foi adotado por ser um dos pacotes de otimização mais utilizados comercialmente.

Na subseção 4.1, apresentamos os resultados obtidos com a heurística. Na subseção 4.2, verificamos a qualidade das soluções em termos de *gap*. E, na subseção 4.3, realizamos uma breve comparação dos resultados com outros autores.

4.1. Resultados

Nosso objetivo, com a metaheurística, é formar carteiras de 5 e 10 ativos, considerando a amostra adotada de 67 ativos. Com base na literatura (principalmente Beasley, Meade e Chang, 2003; e Oh, Kim e Min, 2005), definimos inicialmente o tempo de execução do algoritmo em torno de 5 minutos, com a expectativa de obter respostas com *gap* médio inferior a 10%. Posteriormente, esse tempo foi elevado para em torno de 8 minutos.

Foram realizados quatro testes com alterações de parâmetros (Teste Inicial e Testes 1, 2 e 3), conforme tabela 1. Os parâmetros *Num_Popul* e *Num_Evol* foram definidos de forma que o

Teste Inicial tivesse tempo de execução em torno de 5 minutos (especificamente, tempo entre 4'30'' e 5'30''). Nos Testes 1 e 2, esses dois parâmetros foram alterados conjuntamente de forma que o tempo de execução se mantivesse em 5 minutos; já no Teste 3, o parâmetro *Num_Popul* voltou a receber o valor do Teste Inicial, e o parâmetro *Num_Evol* foi incrementado para termos tempo de execução um pouco maior, em torno de 8 minutos. Os parâmetros *Tax_Mut* e *Num_Mut* foram definidos aleatoriamente no Teste Inicial, e depois alterados nos demais testes, sendo incrementados nos Testes 1 e 2 para formar carteiras de maior diversidade. O parâmetro *Taxa_Crossover* é mantido sempre em 100%. Os parâmetros ϑ e θ foram definidos de forma que fossem iguais em módulo, alterando-se apenas o sinal, para formar o menor intervalo possível entre eles. Os valores foram mantidos em $\vartheta = -0,01$ e $\theta = 0,01$ (ou seja, 1% em módulo) pois, para um intervalo menor do que este entre os parâmetros, algumas carteiras não tiveram solução.

Tabela 1: descrição dos parâmetros dos testes realizados.

Tamanho da Carteira		05 Ativos	10 Ativos	
Teste Inicial	Tempo de execução: 5 minutos	<i>Num_Popul</i>	20	20
		<i>Taxa_Crossover</i>	100%	100%
		<i>Taxa_Mut</i>	85%	85%
		<i>Num_Mut</i>	1	1
		<i>Num_Evol</i>	30	25
Teste 1	Tempo de execução: 5 minutos	<i>Num_Popul</i>	15	15
		<i>Taxa_Crossover</i>	100%	100%
		<i>Taxa_Mut</i>	85%	85%
		<i>Num_Mut</i>	1	1
		<i>Num_Evol</i>	35	30
Teste 2	Tempo de execução: 5 minutos	<i>Num_Popul</i>	10	10
		<i>Taxa_Crossover</i>	100%	100%
		<i>Taxa_Mut</i>	90%	90%
		<i>Num_Mut</i>	2	2
		<i>Num_Evol</i>	50	45
Teste 3	Tempo de execução: 8 minutos	<i>Num_Popul</i>	20	20
		<i>Taxa_Crossover</i>	100%	100%
		<i>Taxa_Mut</i>	80%	80%
		<i>Num_Mut</i>	1	1
		<i>Num_Evol</i>	50	45

Fonte: elaborado pelos autores.

A tabela 2 a seguir demonstra os resultados do Teste Inicial em termos de erro de *tracking* e de *turnover* mensal das carteiras. Conforme esperado, o erro médio aumenta na medida em que o intervalo de rebalanceamento é maior, e o *turnover* mensal médio diminui. Isso evidencia o *trade-off* entre melhor desempenho das carteiras (erro de *tracking*) e menores custos (*turnover* mensal). Ou seja, quanto maior for o intervalo de atualização dos portfólios, os custos tendem a ser menores; porém, como consequência, os erros de *tracking* tendem a aumentar.

Tabela 2: Resultados do Teste Inicial para carteiras de 5 e 10 ativos.

Intervalo fora da amostra	Carteiras de 10 Ativos – Teste Inicial				Carteiras de 05 Ativos – Teste Inicial			
	20	60	120	240	20	60	120	240
Erro de Tracking								
Média	0,345%	0,464%	1,580%	4,760%	0,370%	1,210%	2,017%	2,161%
Mínimo	-2,420%	-2,693%	-1,795%	3,501%	-3,725%	-4,280%	-1,680%	-1,815%
Máximo	3,148%	5,106%	5,701%	5,684%	3,246%	7,389%	7,024%	9,716%
Desvio-padrão	1,262%	2,250%	2,986%	1,130%	1,772%	3,459%	3,367%	6,545%
Turnover Mensal								
Média	36,142%	13,126%	7,728%	4,141%	34,185%	15,598%	8,862%	4,650%
Mínimo	7,982%	6,181%	4,298%	3,578%	1,241%	3,878%	6,040%	3,384%
Máximo	81,263%	20,067%	9,566%	4,783%	78,719%	27,661%	13,146%	5,706%
Desvio-padrão	16,006%	5,587%	1,912%	0,607%	20,699%	7,859%	2,489%	1,175%

Fonte: elaborado pelos autores.

Como ilustração, a figura 3 apresenta a projeção das carteiras de 5 e 10 ativos (com resultados do Teste Inicial) para intervalo fora da amostra de 120 dias úteis (rebalanceamento semestral). Podemos notar a proximidade das curvas das carteiras em relação ao índice, sendo que o portfólio de 10 ativos mantém-se mais próximo (em conformidade com o que se nota na tabela 2, para intervalo de 120 períodos: erro médio de *tracking* e desvio-padrão menores para carteiras de 10 ativos do que para 5 ativos).

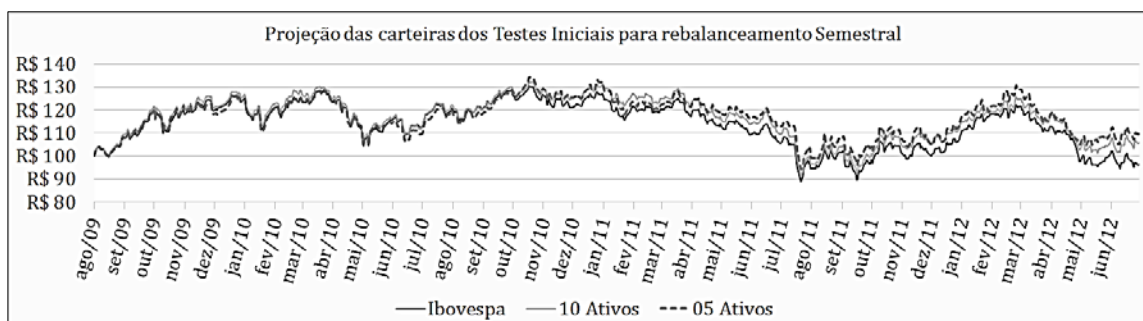


Figura 3: projeção das carteiras para rebalanceamento semestral. Fonte: elaborado pelos autores.

4.2. Verificação do *Gap* dos Resultados

A fim de analisar a qualidade das soluções do algoritmo, optamos por verificar o *gap* destas soluções. Para isso, adotamos o intervalo de rebalanceamento de 20 períodos, em que formamos um total de 36 carteiras, já que podemos formar uma quantidade maior de carteiras para este intervalo do que para os outros intervalos testados.

Para verificação do *gap*, selecionamos aleatoriamente um terço destas 36 carteiras, o que corresponde a 12 carteiras. E, para as 12 carteiras selecionadas (para 5 e 10 ativos), foi executada otimização com o *solver* Cplex (sem uso de heurística) para se obter as respostas ótimas (ou seja, respostas com *gap* de 0%). Devido ao alto tempo de otimização necessário para se obter as soluções ótimas, optamos por fazer essa verificação por “amostragem”, escolhendo 12 carteiras (pois o tempo de processamento para obtenção de cada uma das respostas ótimas ficou, em média, em torno de 5 horas). Com as respostas ótimas, foi calculado o *gap* das soluções do AG para essas 12 carteiras, sendo que $gap = (FO_{AG}/FO_O) - 1$, FO_{AG} é o resultado da função objetivo do AG e FO_O é o resultado da função objetivo da resposta ótima.

A tabela 3 demonstra os resultados de *gap* para cada um dos quatro testes realizados. Dessa forma, podemos constatar que, logo nos Testes Iniciais, as respostas do AG são razoavelmente boas para as carteiras de 10 ativos e, principalmente, muito boas para 5 ativos.

No caso de 10 ativos, temos *gaps* mais elevados; porém, o valor médio permanece em 7,68%, abaixo de 10%, com *gap* mínimo de apenas 3%, sendo que o *gap* ficou acima de 10% para apenas 2 das 12 carteiras testadas (*gap* máximo de 17,58%). No caso de carteiras de 5 ativos, obtivemos *gap* médio de apenas 2,44%; além disso, o algoritmo forneceu solução ótima em 7 dos 12 casos testados. Embora o *gap* máximo seja de 15,1%, trata-se de apenas uma carteira que teve *gap* acima de 10%. Portanto, podemos considerar que os resultados do algoritmo são bastante satisfatórios (pois estamos obtendo respostas ao menos próximas das respostas ótimas com tempo computacional em torno de 5 minutos para problemas que necessitam diversas horas de processamento para obtenção das soluções ótimas sem uso de heurísticas).

Para carteiras de 5 ativos, vemos nos Testes 1 e 2 que as alterações nos parâmetros não influenciaram significativamente as soluções do algoritmo. Para carteiras de 5 ativos, os valores de *gap* médio, máximo e desvio-padrão aumentam nos Testes 1 e 2, o que tende a implicar maiores erros de *tracking*. Assim, notamos uma vantagem nas soluções do Teste Inicial, o que sugere que o uso de uma população maior tende a gerar melhores resultados. No Teste 3, o tempo é aumentado para em torno de 8 minutos. E, com esse pequeno aumento de tempo, para 5 ativos,

verificamos uma melhora considerável das soluções. O *gap* médio diminuiu para apenas 1,05%, com *gap* máximo abaixo de 5%, sendo que a solução ótima foi obtida em 8 das 12 carteiras testadas.

Para carteiras de 10 ativos, o *gap* médio oscila em diferentes sentidos nos Testes 1 e 2; no Teste 1, o *gap* médio aumenta em relação ao Teste Inicial; e no Teste 2, o *gap* médio diminui. Porém, houve aumento do desvio-padrão dos erros. Assim, concluímos que essas alterações de parâmetros não influenciam significativamente os resultados do algoritmo. Já no Teste 3, com tempo em torno de 8 minutos, vemos novamente melhora na qualidade das soluções. O *gap* médio é de apenas 3,77%, com *gap* máximo abaixo de 10%; ou seja, todas as 12 carteiras geradas apresentam *gap* abaixo de 10%, além de o algoritmo ter fornecido solução ótima para 1 das 12 carteiras testadas. Assim, vemos novamente que um pequeno aumento no tempo de processamento é suficiente para causar um impacto considerável nas soluções.

Tabela 3: Valores de *gap* para carteiras de 5 e 10 ativos (para rebalanceamento mensal).

Testes	Gap para Carteiras de 10 Ativos				Gap para Carteiras de 5 Ativos			
	Teste Inicial ⁽¹⁾	Teste 1 ⁽²⁾	Teste 2 ⁽³⁾	Teste 3 ⁽⁴⁾	Teste Inicial ⁽⁵⁾	Teste 1 ⁽⁶⁾	Teste 2 ⁽⁷⁾	Teste 3 ⁽⁸⁾
Média	7,68%	7,95%	6,61%	3,78%	2,44%	4,02%	4,70%	1,05%
Mínimo	3,05%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Máximo	17,58%	19,71%	17,02%	7,86%	15,13%	23,02%	15,66%	4,73%
Desvio-padrão	4,05%	5,63%	6,45%	2,21%	4,44%	6,72%	6,34%	1,69%

⁽¹⁾Teste Inicial: algoritmo forneceu *gap* acima de 10% para 2 das 12 carteiras testadas.

⁽²⁾Teste 1: algoritmo forneceu solução ótima para 1 das 12 carteiras testadas. Para 4 carteiras, o *gap* ficou acima de 10%.

⁽³⁾Teste 2: algoritmo forneceu solução ótima para 2 das 12 carteiras testadas. Para 3 carteiras, o *gap* ficou acima de 10%.

⁽⁴⁾Teste 3: algoritmo forneceu solução ótima para 1 das 12 carteiras testadas.

⁽⁵⁾Teste Inicial: algoritmo forneceu solução ótima para 7 das 12 carteiras testadas; e 1 carteira ficou com *gap* acima de 10%.

⁽⁶⁾Teste 1: algoritmo forneceu solução ótima para 6 das 12 carteiras testadas. Para 2 carteiras, o *gap* ficou acima de 10%.

⁽⁷⁾Teste 2: algoritmo forneceu solução ótima para 7 das 12 carteiras testadas. Para 3 carteiras, o *gap* ficou acima de 10%.

⁽⁸⁾Teste 3: algoritmo forneceu solução ótima para 8 das 12 carteiras testadas.

Fonte: elaborado pelos autores.

Como conclusão, verificamos que o algoritmo cumpre os objetivos propostos. Com a heurística, foi possível formar carteiras com número reduzido de ativos com tempo baixo de processamento e valores médios de *gap* abaixo de 10%; e, com tempo um pouco maior, em torno de 8 minutos, obtivemos *gap* médio abaixo de 5%, sendo 8 minutos um tempo baixo se considerarmos que situações reais exigiriam a execução do algoritmo por diversas vezes em apenas um dia, a fim de se testar diferentes cenários antes da definição da nova carteira, especialmente em situações de movimentos bruscos do mercado. Dessa forma, podemos ter um alto grau de confiança de que estamos obtendo soluções, ao menos, bastante próximas das soluções ótimas.

4.3. Comparação com Outros Autores

Pela análise da literatura, não foram localizados estudos que usaram a mesma base de dados deste artigo (utilizando o índice Ibovespa como referência). Assim, a comparação com resultados de outros autores torna-se um pouco mais difícil. Apesar disso, realizamos uma breve comparação com Beasley, Meade e Chang (2003) e Guastaroba e Speranza (2012) – pois estes dois estudos utilizaram, entre si, os mesmos índices de referência e bases de dados iguais.

A tabela 4 demonstra de forma sucinta os resultados destes dois estudos e os resultados

que obtivemos para o índice Ibovespa. Para fazer a comparação, usamos os valores de erro de *tracking* para cada tipo de carteira formada (5 e 10 ativos) e o respectivo valor de Percentual. O cálculo do erro de *tracking* (TE), neste caso, corresponde à diferença entre a soma dos quadrados dos desvios dos retornos de cada carteira em relação ao índice (considerando os retornos diários do intervalo fora da amostra). O valor de Percentual é obtido pela fórmula $(TE/Desv_{R_t}) * 100$, sendo $Desv_{R_t}$ o desvio padrão dos retornos diários do índice no intervalo fora da amostra. No caso dos “Resultados para Índice Ibovespa”, utilizamos os resultados das carteiras com rebalanceamento semestral (com resultados do Teste Inicial do AG); considerando que, com esse rebalanceamento, podemos formar um total de 6 carteiras, os valores de erro de *tracking* e de Percentual correspondem à média do erro e média do Percentual dessas 6 carteiras. Para fazermos essa comparação, tomamos como referência a relação C/N , em que C é o tamanho das carteiras e N é o tamanho das amostras.

Assim, podemos perceber inicialmente que o erro para o Índice Ibovespa aumenta consideravelmente de 10 para 5 ativos. Para 10 ativos, podemos comparar os resultados do Ibovespa com os resultados de Beasley, Meade e Chang (2003) e Guastaroba e Speranza (2012) para carteiras de relação $C/N = 0,12$. Nesse caso, temos erro de *tracking* menor e um percentual um pouco reduzido, o que sugere que a volatilidade do Ibovespa é superior à volatilidade do índice usado por Beasley, Meade e Chang (2003) e Guastaroba e Speranza (2012).

Já no caso de carteiras de 5 ativos, poderíamos comparar os resultados do Ibovespa com os resultados de Beasley, Meade e Chang (2003) e Guastaroba e Speranza (2012) para carteiras com $C/N = 0,10$. Nesse ponto, notamos um erro de *tracking* bastante elevado para o Ibovespa, e por consequência alto percentual. Considerando que a análise de *gap* das soluções do algoritmo nos sugere que essas soluções estão ao menos próximas das respostas ótimas, podemos concluir sobre a impossibilidade de se realizar o *tracking* do Ibovespa com carteiras de apenas 5 ativos (ao menos no caso de períodos maiores de rebalanceamento).

Em suma, conforme citado anteriormente, o uso de uma diferente base de dados torna essa comparação um pouco difícil de ser feita. Apesar disso, foi possível constatar que os resultados obtidos para o Ibovespa estão, ao menos, próximos aos de outros autores, especialmente para carteiras de 10 ativos.

Tabela 4: Comparação com Beasley, Meade e Chang (2003) e Guastaroba e Speranza (2012).

Beasley, Meade e Chang (2003)				
N*	C**	C/N***	Tracking error	Percentual
85	10	0,12	0,2049%	10,091
98	10	0,10	0,1032%	6,805
225	10	0,04	0,0821%	2,890
Guastaroba e Speranza (2012)				
N	C	C/N	Tracking error	Percentual
85	10	0,12	0,2048%	10,090
98	10	0,10	0,1032%	6,805
225	10	0,04	0,0821%	2,890
Resultados para Índice Ibovespa				
N	C	C/N	Tracking error	Percentual
67	10	0,15	0,0176%	3,241
67	5	0,07	0,1913%	13,719

*N = Tamanho da amostra; **C = Tamanho da carteira;

***C/N = Relação entre carteira e amostra.

Fonte: elaborado pelos autores.

5. Conclusões

Neste estudo, aplicamos ao mercado brasileiro um modelo de *index tracking* com controle do número de ativos. Considerando a forte restrição na quantidade de ativos para

formação das carteiras, usamos o método de algoritmos genéticos, pois o problema possui alta complexidade computacional e não pode ser resolvido em poucos minutos sem uso de heurísticas.

Com aplicação da metaheurística, foram atingidos os objetivos centrais de se formar carteiras de 5 e 10 ativos com *gap* das soluções abaixo de 10% e tempo de processamento em torno de 5 minutos. E, ao aumentarmos esse tempo para em torno de 8 minutos, tornou-se possível obter respostas com *gap* médio abaixo de 5% tanto para carteiras de 5 ativos quanto para 10 ativos, obtendo-se assim um forte indicativo de que o algoritmo forneceu soluções de boa qualidade.

Por fim, comparações dos resultados com outros autores, especificamente Beasley, Meade e Chang (2003) e Guastaroba e Speranza (2012), demonstraram que nossa aplicação de *index tracking* para o mercado brasileiro forneceu soluções similares às aplicações para outros mercados, especialmente para formação de carteiras de 10 ativos (sendo necessário considerar a certa superficialidade existente nessa comparação, dado o uso de diferentes amostras de dados). Já no caso de carteiras de 5 ativos, obtivemos soluções de erro bastante alto se comparado com carteiras similares em Beasley, Meade e Chang (2003) e Guastaroba e Speranza (2012); porém, dado o baixo *gap* dessas soluções, podemos concluir que realizar o *tracking* do Ibovespa com apenas 5 ativos não é possível (ao menos para períodos maiores de rebalanceamento).

Como sugestões de estudos futuros, seria indicado basicamente a aplicação desta heurística com amostras de dados de maior quantidade de ativos, para que assim seja possível realizar comparações diretas com outros autores. Além disso, poderiam ser incluídas mais restrições no modelo, para controle de *turnover* e de pesos mínimo e máximo dos ativos, e novas execuções da heurística com maior variação dos parâmetros.

Referências

- Barro, Diana & Canestrelli, Elio** (2009). Tracking error: a multistage portfolio model. *Annals of Operations Research*, 165:47-66.
- Beasley, John E., Meade, Nigel & Chang, Tien-Jye** (2003). An evolutionary heuristic for the index tracking problem. *European Journal of Operational Research*, 148:621-643.
- Borges, Elaine C., Eid Jr., Willian & Yoshinaga, Cláudia E.** (2012). Exchange Traded Funds versus Fundos Indexados no Brasil. *Revista de Finanças Aplicadas*, 1:1-15.
- Caldeira, João F. & Portugal, Marcelo** (2010). Estratégia Long-Short, Neutra ao Mercado, e Index Tracking Baseadas em Portfólios Cointegrados. *Revista Brasileira de Finanças*, 8:469-504.
- Canakgoz, Nilgun A. & Beasley, John E.** (2009). Mixed-integer programming approaches for index tracking and enhanced indexation. *European Journal of Operational Research*, 196:384-399.
- Coleman, Thomas F., Li, Yuying & Henniger, Jay** (2006). Minimizing tracking error while restricting the number of assets. *Journal of Risk*, 8:33-56.
- Dunis, Christian L. & Ho, Richard** (2005). Cointegration Portfolios of European Equities for Index Tracking and Market Neutral Strategies. *Journal of Asset Management*, 6:33-52.
- Fama, Eugene** (1970). Efficient Capital Markets: A Review of Theory and Empirical Work. *Journal of Finance*, 25:383-417.
- Fama, Eugene & French, Kenneth** (2010). Luck versus Skill in the Cross-Section of Mutual Fund Returns. *The Journal of Finance*, 65:1915-1947.
- Filomena, Tiago P. & Lejeune, Miguel A.** (2014). Warm-Start Heuristic for Stochastic Portfolio Optimization with Fixed and Proportional Transaction Costs. *Journal of Optimization Theory and Applications*, 161:308-329.
- Frino, Alex & Gallagher, David R.** (2001). Tracking S&P 500 Index Funds. *Journal of Portfolio Management*, 28:44-55.
- Gaivoronski, Alexei A., Krylov, Sergiy & van der Wijst, Nico** (2005). Optimal portfolio selection and dynamic benchmark tracking. *European Journal of Operational Research*, 163:115-131.
- Guastaroba, Gianfranco & Speranza, Maria Grazia** (2012). Kernel Search: An application to the index tracking problem. *European Journal of Operational Research*, 217:54-68.
- Jansen, Roel & van Dijk, Ronald** (2002). Optimal Benchmark Tracking with Small Portfolios. *The Journal of Portfolio Management*, 28:33-39.
- Jeurissen, Roland & van den Berg, Jan** (2008). Optimized index tracking using a hybrid genetic

- algorithm. IEEE Congress on Evolutionary Computation, Hong Kong.
- Maringer, Dietmar & Oyewumi, Olufemi** (2007). Index Tracking with Constrained Portfolios. *Intelligent Systems in Accounting, Finance and Management*, 15:57-71.
- Markowitz, Harry** (1952). Portfolio Selection. *Journal of Finance*, 7:77-91.
- Oh, Kyong Joo, Kim, Tae Yoon & Min, Sungky** (2005). Using genetic algorithm to support portfolio optimization for index fund management. *Expert Systems with Applications*, 28:371-379.
- Roll, Richard** (1992). A Mean/Variance Analysis of Tracking Error. *Journal of Portfolio Management*, 18:12-22.
- Sant'Anna, Leonardo R.; Filomena, Tiago P.; Borenstein, Denis** (2013). Estratégia de Index Tracking Aplicada ao Índice Ibovespa para o Período de 2009 a 2012. In: 13º Encontro Brasileiro de Finanças (EBFin), 13., Rio de Janeiro, 2013. Anais. Rio de Janeiro: Sociedade Brasileira de Finanças.
- Scozzari, Andrea, Tardella, Fabio, Paterlini, Sandra & Krink, Thiemo** (2012). Disponível em <http://ssrn.com/abstract=2066672>. Acesso em: 27/4/2014.