



UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

OTIMIZAÇÃO HEURÍSTICA APLICADA À SELEÇÃO DE PORTFÓLIO DE  
PROJETOS DE SOFTWARE

Fábio Vitorino Figueiredo

**Orientador**

Márcio de Oliveira Barros

RIO DE JANEIRO, RJ – BRASIL  
SETEMBRO DE 2011

OTIMIZAÇÃO HEURÍSTICA APLICADA À SELEÇÃO DE PORTFÓLIO DE  
PROJETOS DE SOFTWARE

Fábio Vitorino Figueiredo

DISSERTAÇÃO APRESENTADA COMO REQUISITO PARCIAL PARA  
OBTENÇÃO DO TÍTULO DE MESTRE PELO PROGRAMA DE PÓS-  
GRADUAÇÃO EM INFORMÁTICA DA UNIVERSIDADE FEDERAL DO ESTADO  
DO RIO DE JANEIRO (UNIRIO). APROVADA PELA COMISSÃO  
EXAMINADORA ABAIXO ASSINADA.

Aprovada por:



Mircio de Oliveira Barros, D.Sc. – UNIRIO



Renata Mendes de Araújo, D.Sc. – UNIRIO



Ana Regina Cavalcanti da Rocha, D.Sc. – COPPE/UFRJ

RIO DE JANEIRO, RJ – BRASIL

SETEMBRO DE 2011

F475 Figueiredo, Fábio Vitorino.  
Otimização heurística aplicada à seleção de portfólio de projetos de software / Fábio Vitorino Figueiredo, 2011.  
79f.

Orientador: Márcio de Oliveira Barros.  
Dissertação (Mestrado em Informática) – Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2011.

1. Engenharia de software. 2. Portfólios (Projeto de software) – Seleção. 3. Projeto de software – Seleção. 4. Otimização combinatória. I. Barros, Márcio de Oliveira. II. Universidade Federal do Estado do Rio de Janeiro (2003-). Centro de Ciências Exatas e Tecnologia. Curso de Mestrado em Informática. III. Título.

CDD – 005.1

*À minha querida avó.  
À minha admirável mãe.  
À minha amada esposa.*

## **AGRADECIMENTOS**

Ao Deus criador do universo, pai de nosso Senhor e Salvador Jesus Cristo, que me concedeu a benção de chegar até aqui e concluir este trabalho, vencendo as várias dificuldades e obstáculos encontrados no caminho.

À minha avó Célia pelos conselhos que enriqueceram o meu discernimento.

À minha mãe Vera, pelos sacrifícios feitos no passado e pelo apoio incondicional dado a cada etapa desta pesquisa.

À minha digníssima esposa Nathália, a quem agradeço as orações que me fizeram mais forte.

A todos os meus familiares que estiveram torcendo pelo meu sucesso.

Ao meu orientador, professor Márcio Barros, que executou com excelência e grande maestria o trabalho de nortear cada atividade ao longo desse percurso e pelo precioso tempo doado a este trabalho.

À professora Renata M. de Araújo, coordenadora do Núcleo de Pesquisa, Prática e Tecnologia (NP2Tec) da UNIRIO e à professora Ana Regina C. da Rocha, por aceitarem o convite para composição da banca avaliadora desta dissertação de mestrado.

Aos demais professores, que por ocasião dos seminários discentes e aulas, nos brindaram com suas valiosas observações e críticas construtivas.

A todos os funcionários da secretaria, pela extrema competência e atenção às nossas questões administrativas.

E aos amigos do mestrado, que tornaram todo o processo menos árduo, com as piadas, dicas e caronas.

FIGUEIREDO, Fábio Vitorino. **Otimização Heurística Aplicada à Seleção de Portfólio de Projetos de Software**. UNIRIO, 2011.  
079 páginas. Dissertação de Mestrado.  
Departamento de Informática Aplicada, UNIRIO.

## RESUMO

Este trabalho de pesquisa propõe uma heurística que permite que uma técnica de apoio à seleção de portfólio de projetos de software seja aplicada em cenários com grande número de projetos candidatos. Enquanto a técnica original se baseia na análise de todas as combinações possíveis dos projetos para posterior seleção da melhor combinação, propomos o uso de um algoritmo genético para explorar parcialmente o conjunto destas alternativas e selecionar uma combinação viável sob o aspecto da relação risco/retorno do portfólio. A pesquisa apresenta um estudo experimental que compara a heurística proposta com uma técnica de busca local multi-partida (*Hill Climbing Random Restart*) e uma técnica de busca não-sistemática (*Random Search*). Os resultados do estudo confirmam que a técnica de seleção de portfólio de projetos de software pode ser utilizada em larga escala e que a heurística proposta apresenta melhores resultados que as outras estratégias de busca.

**Palavras-chave:** Seleção de Portfólios; Seleção de Projetos; Otimização Combinatória

## ABSTRACT

This research proposes a heuristic approach to allow a software project portfolio selection technique to be used on scenarios with a large number of candidate projects to compose the portfolio. While the original technique is based on analyzing all possible project combinations for selecting the best one, we propose using a genetic algorithm to partially explore this search space and select an efficient project combination. We present an experimental study that compares the proposed heuristic with a local search technique (*Hill Climbing Random Restart*) and a non-systematic search technique (*Random Search*). Its results confirm that the project portfolio technique can be used in large scale and that the proposed heuristic presents better results than the other search strategies selected to be part of the experimental study.

**Keywords:** Portfolio Selection; Project Selection; Combinatorial Optimization

## SUMÁRIO

<b>INTRODUÇÃO .....</b>	<b>1</b>
1.1. CONSIDERAÇÕES INICIAIS.....	1
1.2. MOTIVAÇÃO .....	2
1.3. OBJETIVOS.....	5
1.4. METODOLOGIA DE PESQUISA.....	6
1.5. ESTRUTURA DO TEXTO DA DISSERTAÇÃO.....	6
<b>REVISÃO BIBLIOGRÁFICA .....</b>	<b>8</b>
2.1. CONSIDERAÇÕES INICIAIS .....	8
2.2. GERENCIAMENTO DE PORTFÓLIO.....	9
2.3. MODERNA TEORIA DO PORTFÓLIO .....	10
2.4. TÉCNICAS DE OTIMIZAÇÃO PARA SELEÇÃO DE UM PORTFÓLIO .....	12
2.4.1. Busca Local para Seleção de um Portfólio.....	12
2.4.2. Busca Heurística para Seleção de um Portfólio.....	14
2.4.3. Busca Heurística para Seleção de um Portfólio em Estágios .....	15
2.4.4. Busca Heurística para Seleção de um Portfólio com Restrição de Lote ....	19
2.4.5. Busca Heurística de um Portfólio usando Diferentes Medidas de Risco ...	24
2.5. CONCLUSÃO .....	25
<b>PROPOSTA DE SOLUÇÃO PARA A TÉCNICA DE SELEÇÃO DE PORTFÓLIO DE PROJETOS .....</b>	<b>26</b>
3.1. CONSIDERAÇÕES INICIAIS .....	26
3.2. BUSCA HEURÍSTICA COM ALGORITMOS GENÉTICOS .....	27
3.3. O PROBLEMA DE SELEÇÃO DO PORTFÓLIO DE PROJETOS .....	29
3.4. FORMALIZAÇÃO DO PROBLEMA DE SELEÇÃO DO PORTFÓLIO DE PROJETOS.....	32
3.5. PROPOSTA DE SOLUÇÃO.....	36
3.6. CONCLUSÃO .....	37
<b>AVALIAÇÃO DO MÉTODO PROPOSTO .....</b>	<b>39</b>
4.1. CONSIDERAÇÕES INICIAIS .....	39
4.2. QUESTÕES DE PESQUISA .....	40
4.3. INSTÂNCIAS UTILIZADAS NO ESTUDO .....	41
4.4. ALGORITMOS UTILIZADOS NO ESTUDO.....	44
4.5. ANÁLISE DOS RESULTADOS .....	46
4.5.1. Resultados do Estudo Experimental sobre Instâncias Aleatórias.....	47
4.5.2. Resultados do Estudo Experimental sobre Instâncias do Mundo Real .....	54
4.6. AMEAÇAS À VALIDADE DO ESTUDO .....	57
4.7. CONCLUSÃO .....	59
<b>CONCLUSÕES.....</b>	<b>61</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>65</b>

## ÍNDICE DE FIGURAS

Figura 2.1 – Gráfico da Fronteira Eficiente .....	11
Figura 2.2 – Esquema de representação de um investimento .....	16
Figura 2.3 – Gráfico representativo da fronteira eficiente .....	21
Figura 3.1 – Uma representação dos elementos envolvidos no problema .....	31
Figura 4.1 – Custo computacional medido nas instâncias com 25 projetos .....	50
Figura 4.2 – IDP( $P_z$ ) nas instâncias com 25 projetos .....	50
Figura 4.3 – Custo computacional medido nas instâncias com 50 projetos .....	51
Figura 4.4 – IDP( $P_z$ ) nas instâncias com 50 projetos .....	51
Figura 4.5 – Custo computacional medido nas instâncias com 75 projetos .....	52
Figura 4.6 – IDP( $P_z$ ) nas instâncias com 75 projetos .....	52
Figura 4.7 – Custo computacional medido nas instâncias com 100 projetos .....	53
Figura 4.8 – IDP( $P_z$ ) nas instâncias com 100 projetos .....	53
Figura 4.9 – Custo computacional medido nas instâncias do mundo real .....	56
Figura 4.10 – IDP( $P_z$ ) nas instâncias do mundo real .....	56

## ÍNDICE DE TABELAS

Tabela 1.1 – Resultados obtidos com a busca exaustiva .....	4
Tabela 1.2 – Estimativa de custo computacional com a busca exaustiva .....	4
Tabela 2.1 – Representação dos indicadores usados por LAI et al. (2006) .....	16
Tabela 4.1 – Custo computacional despendido nas instâncias aleatórias .....	47
Tabela 4.2 – Índices de desempenho dos portfólios obtidos nas instâncias aleatórias ..	48
Tabela 4.3 – Custo computacional medido nas instâncias do mundo real .....	54
Tabela 4.4 – Índice de desempenho do portfólio das instâncias do mundo real .....	55

## INTRODUÇÃO

---

### 1.1. Considerações Iniciais

A evolução tecnológica, a expansão dos negócios online e a sombra constante de uma economia incerta contribuem para que as organizações procurem aprimorar suas estratégias e métodos para melhoria de seus produtos, processos e serviços, de modo que possam alcançar vantagens competitivas no mercado. Sob esse cenário complexo de globalização e competição, são comuns as constantes tomadas de decisão acerca das diversas tarefas executadas internamente, sabendo-se que a decisão tomada pode gerar lucros ou prejuízos à organização, dependendo dos resultados ocasionados por ela. Além disso, a escolha de algumas dessas tarefas internas em detrimento de outras ainda pode gerar conflitos e competições entre os setores. Tudo isso faz com que a decisão seja tomada com cautela pelos gestores, que por sua vez costumam adotar técnicas que os auxiliem a tomar as decisões certas, sob o aspecto de serem as decisões mais favoráveis à organização nos contextos financeiro, social, ambiental e tecnológico.

Segundo o Guia Geral do MR MPS (SOFTEX, 2011), a Engenharia de Software, como diversas áreas de conhecimento, também requer o uso de técnicas gerenciais, pois decisões precisam ser tomadas ao longo de todo o processo de desenvolvimento e evolução dos sistemas. RUHE (2003) comenta que a tomada de decisões afeta significativamente todos os estágios do ciclo de vida de um projeto e que processos e sistemas de apoio à decisão são fundamentais para aumentar a eficiência, a qualidade e a relação custo/benefício dos projetos.

Dentre as diversas decisões que devem ser tomadas por uma empresa desenvolvedora de software encontra-se a decisão de quais projetos serão executados no período, levando-se em consideração o orçamento disponível para este fim. O conjunto desses

projetos escolhidos formará o portfólio da organização, conceito que será introduzido no Capítulo 2 desta dissertação.

Atualmente, o gerenciamento do portfólio de projetos, além de ser uma atividade comum (ARCHER e GHASEMZADEH, 1999), é de suma importância para a estratégia de negócio de uma organização (PADOVANI, 2007). Contudo, a implementação de uma gestão eficiente do portfólio de projetos não é uma tarefa simples, na medida em que envolve diversos fatores potencialmente conflitantes, tais como recursos, tecnologias, prazos, incertezas, tomada de decisão, estimativas de valores, entre outros.

## **1.2. Motivação**

Os escassos recursos da organização geralmente não permitem a execução de todos os projetos apresentados ao gestor, sendo necessário fazer uma seleção dentre esses projetos para que o portfólio seja formado de acordo com os recursos disponíveis na organização, de forma a maximizar os lucros e minimizar os riscos (COOPER et al., 2001). Portfólios mal formados podem incluir projetos que não cumpram o objetivo para o qual foram criados e isso geralmente causa desperdício de tempo, trabalho e dinheiro na organização.

Recentemente, COSTA (2011) propôs uma técnica de apoio à seleção de portfólio de projetos de software baseada na Moderna Teoria do Portfólio (MARKOWITZ, 1952). Essa técnica permite que o gestor identifique os portfólios que possuem os maiores índices de eficiência e que possam ser compostos a partir de um conjunto de projetos candidatos. O índice de eficiência do portfólio é calculado dividindo-se o retorno esperado do portfólio por seu risco e será chamado daqui por diante de IDP (Índice de Desempenho do Portfólio). O retorno do portfólio é estimado a partir dos retornos esperados para os projetos, enquanto o risco do portfólio é calculado de acordo com a exposição individual de cada projeto a uma série de incertezas relevantes, que podem ser ameaças ou oportunidades, identificadas pelo gestor. Com base nas relações existentes entre os projetos candidatos e estas incertezas, calcula-se o grau de dependência entre os projetos, que permite considerar a diversificação dos riscos no portfólio.

Os estudos realizados por COSTA (2011) demonstram que a técnica proposta é capaz de maximizar a relação risco/retorno do ponto de vista do gestor, quando

comparada com outras técnicas de seleção de portfólio propostas na literatura técnica e em uso por profissionais da área no contexto da seleção de portfólio de projetos de software, conforme pode ser visto nas comparações feitas em seu trabalho.

Os fatores descritos acima contribuem para que os resultados apresentados pela técnica proposta por COSTA (2011) sejam confiáveis e, por conta disso, ela foi escolhida como base deste trabalho. No entanto, a técnica possui complexidade computacional exponencial com relação ao número de projetos candidatos ao portfólio. Isto ocorre porque a técnica se baseia na análise de todas as possíveis combinações entre esses projetos, ou seja, são construídos e avaliados todos os portfólios que podem ser formados por subconjuntos dos projetos candidatos e esse número afeta o tempo de execução da técnica, inviabilizando seu uso em larga escala. Por exemplo, ao utilizar a técnica para um cenário com cinco projetos candidatos, um gestor terá que avaliar 32 portfólios. No entanto, se houver 40 projetos candidatos, o número de possíveis combinações passa de um trilhão. Neste cenário, a técnica não pode ser aplicada em tempo aceitável<sup>1</sup>.

Neste cenário, o presente trabalho de pesquisa apresenta uma abordagem para complementar o trabalho de COSTA (2011) nos casos com elevado número de projetos. Para demonstrar a inviabilidade de execução da técnica original em cenários com grande número de projetos, implementamos um algoritmo de busca exaustiva para ser aplicado com a estratégia contida na técnica e o executamos sobre três cenários hipotéticos (13, 14 e 15 projetos, cada um contendo 10 riscos). O tempo de execução da técnica nestes cenários é apresentado na Tabela 1.1, cuja notação "xP yR" representa a quantidade de projetos (x) e riscos (y) existentes no cenário.

O esforço computacional despendido pela busca exaustiva foi estimado através de uma regra de três simples, aplicada sobre a relação *Nº de Avaliações x Tempo de Processamento*, onde o *Nº de Avaliações* representa a quantidade de possíveis combinações que podem ser feitas entre os projetos candidatos ao portfólio e o Tempo de Processamento corresponde ao tempo, medido em segundos, exigido pelo algoritmo até chegar ao resultado. Como todos os possíveis portfólios devem ser avaliados em todos os possíveis cenários de risco, o número de avaliações executadas é obtido

---

<sup>1</sup> Tempo que um gestor ou tomador de decisão está disposto a esperar por uma solução, do ponto de vista prático.

multiplicando-se a quantidade de combinações possíveis de projetos pela quantidade de combinações possíveis de riscos, ou seja,  $(2^N * 2^R)$ , onde N corresponde ao número de projetos candidatos ao portfólio e R corresponde ao número de riscos identificados. A tabela 1.1 apresenta o número de avaliações executadas e o tempo de processamento do algoritmo de busca exaustiva para analisar todas as possíveis combinações de projetos e riscos para três cenários hipotéticos.

**Tabela 1.1 – Resultados obtidos com a busca exaustiva**

<i>Projetos / Riscos</i>	<i>Nº de Avaliações</i>	<i>Tempo de Processamento</i>
13P / 10R	8.388.608	9 segundos
14P / 10R	16.777.216	19 segundos
15P / 10R	33.554.432	44 segundos

A primeira linha da tabela 1.1 mostra que cerca de 930 mil avaliações são realizadas por segundo no computador utilizado neste estudo. Assumindo que esta relação linear se mantém à medida que o tamanho da instância cresce, podemos estimar o tempo necessário para as instâncias de 14 projetos (18 segundos) e 15 projetos (36 segundos). Como pode ser observado nas duas últimas linhas da tabela, o tempo efetivamente necessário para calcular estas instâncias é superior ao estimado (provavelmente por conta da impossibilidade de manter todas as informações na memória principal do computador, exigindo a execução de *swaps* pelo sistema operacional), o que indica que a relação linear é um limite inferior para a estimativa de tempo de processamento.

Escalando essa projeção para instâncias maiores e considerando o tempo mínimo de processamento necessário para a busca exaustiva como o indicado pela relação linear, obtemos os valores apresentados na tabela 1.2 para situações variando entre 25 e 100 projetos candidatos.

**Tabela 1.2 – Estimativa de custo computacional com a busca exaustiva**

<i>Projetos / Riscos</i>	<i>Nº de Avaliações</i>	<i>Estimativa</i>
25P / 10R	3,43 E+10	10 horas
50P / 10R	1,12 E+15	39768 anos
75P / 10R	3,77 E+22	1,33E+10 séculos
100P / 10R	1,26 E+30	4,47E+17 séculos

Estes dados confirmam a necessidade de uma estratégia de busca mais eficiente que a busca exaustiva para que os conceitos descritos na técnica proposta por COSTA (2011)

possam ser utilizados em cenários com um grande número de projetos candidatos a fazer parte do portfólio.

Cabe ressaltar que o sucesso da técnica proposta por COSTA (2011) aplicada com o método de otimização proposto nesta dissertação está sujeito ao nível de maturidade da organização em gerência de projetos e portfólios, no que se refere aos fatores: (a) grau de confiabilidade dos valores estimados para o retorno previsto de cada projeto; (b) cálculo, no mínimo aproximado, do custo de cada projeto; (c) estimativa correta da probabilidade de ocorrência de cada incerteza identificada no cenário; e (d) identificação dos possíveis impactos gerados por cada incerteza em cada projeto.

### **1.3. Objetivos**

O objetivo geral desta dissertação é o desenvolvimento de um modelo para tornar a técnica proposta por COSTA (2011) aplicável em escala quanto ao número de projetos candidatos ao portfólio, através do uso de uma busca heurística baseada em algoritmos genéticos. Neste sentido, é necessária a modelagem do problema de seleção do portfólio de projetos de software como um problema de otimização, para o qual as técnicas de busca possam encontrar soluções eficientes.

Esperamos responder à hipótese de que se faz necessário o uso de uma heurística para aplicar a técnica proposta por COSTA (2011), na busca de soluções viáveis em cenários com elevado número de projetos candidatos a compor o portfólio. Para alcançarmos o objetivo geral, temos os seguintes objetivos específicos:

- (a) Modelagem do problema de seleção do portfólio de projetos de software como um problema de otimização;
- (b) Desenvolvimento de um método heurístico de busca, na forma de um algoritmo genético, que possa procurar soluções viáveis dentro do espaço de soluções alternativas geradas por uma instância do problema, sem que haja a análise de todas as soluções alternativas;
- (c) Realização de um estudo experimental que utilize outros dois métodos de busca para comparação de seus resultados com os resultados do método heurístico, para avaliar e validar os resultados obtidos, além de verificar se a busca heurística é efetivamente necessária e indicada para resolver este tipo de problema, no contexto da seleção de

um portfólio através da técnica descrita em COSTA (2011), sob o aspecto da maximização da relação risco/retorno do portfólio.

Assim como no trabalho de COSTA (2011), a utilização da estratégia contida nessa técnica trata apenas da maximização do retorno e minimização do risco do portfólio, não levando em consideração os critérios que apontam os projetos mais alinhados aos objetivos estratégicos da organização e, portanto, deve ser aplicada nos projetos que já tiverem passado por um crivo inicial no que diz respeito a esses critérios, tornando a implementação deste modelo apenas uma ferramenta auxiliar de apoio à decisão.

#### **1.4. Metodologia de Pesquisa**

Segundo LACHTERMACHER (2004), esta pesquisa pode ser classificada como tecnológica quanto à sua natureza, pois visa gerar um novo processo de utilização da técnica proposta por COSTA (2011) em cenários onde haja um grande número de projetos candidatos a compor o portfólio da organização. Com relação aos objetivos da pesquisa científica, pode ser considerada exploratória, pois buscamos conhecer os fatos e fenômenos relacionados ao tema através de um levantamento bibliográfico e entrevistas com profissionais da área, dentre eles, o próprio autor da técnica estudada. Quanto aos procedimentos, esta pesquisa pode ser considerada como experimental, uma vez que a estratégia descrita na técnica estudada foi reproduzida de forma controlada em instâncias com elevado número de projetos, para que comparássemos os resultados entre o método proposto e outros dois métodos de busca e avaliássemos a aplicabilidade da proposta. Esse estudo experimental buscou verificar, de forma quantitativa, se o modelo proposto consegue selecionar portfólios considerados melhores que os portfólios selecionados pelas outras técnicas de busca utilizadas no estudo, sob o aspecto do maior valor obtido para o índice que representa a relação risco/retorno do portfólio, em cenários onde o número de projetos aumenta significativamente.

#### **1.5. Estrutura do texto da dissertação**

Este trabalho é composto por cinco Capítulos, incluindo esta introdução.

No Capítulo 2 apresentamos uma revisão da literatura técnica, onde abordamos inicialmente o conceito de projetos e portfólios, para, em seguida, discorrermos sobre o

gerenciamento de portfólios. Após esses conceitos, fazemos uma breve apresentação sobre algumas técnicas encontradas na literatura para seleção de portfólio.

No Capítulo 3 abordamos buscas heurísticas com algoritmos genéticos, para então apresentarmos a técnica de apoio à seleção de portfólio de projetos de software baseado na Moderna Teoria do Portfólio (COSTA, 2011), escolhida para fazer parte deste estudo. A seguir, apresentamos formalmente o problema da seleção de projetos para composição de um portfólio e a proposta para solução desse problema.

No Capítulo 4 descrevemos a configuração e a condução do estudo experimental que foi feito para validar a proposta de solução apresentada no Capítulo 3, fazendo uma análise dos resultados observados e discutindo as ameaças que podem afetar a validade desse estudo.

Por fim, no Capítulo 5 apresentamos as considerações finais sobre o estudo, enfocando suas contribuições, limitações e propostas para trabalhos futuros.

### REVISÃO BIBLIOGRÁFICA

---

#### 2.1. Considerações iniciais

Projetos são esforços temporários, empreendidos com um objetivo pré-estabelecido que, normalmente, está ligado aos objetivos estratégicos da organização (PMI, 2008). Apresentam níveis diferenciados de risco e de retorno e, por competirem pelos mesmos recursos, se tornam difíceis de selecionar e gerenciar. Com o intuito de tratar o problema de gerir múltiplos projetos surgiu o conceito de portfólio no início dos anos 50 (MARKOWITZ, 1952).

Segundo o PMI (2008), um portfólio é uma coleção de projetos, programas, portfólios e outros trabalhos que são agrupados para facilitar a gerência efetiva do trabalho e o alcance dos objetivos estratégicos de negócio. A gerência de um portfólio inclui a identificação, categorização, avaliação, seleção, priorização, autorização e controle dos projetos, programas e outros trabalhos relacionados, baseado em seu alinhamento e contribuição aos objetivos e estratégias da organização.

Manter um portfólio faz parte de uma estratégia de diversificação com o intuito de diminuir riscos e maximizar lucros (COOPER et al., 2001), além de trazer uma série de outros benefícios para as organizações (EPM CONNECT, 2007): (a) auxilia na seleção e priorização de idéias e projetos candidatos para execução; (b) permite que os investimentos se mantenham alinhados aos objetivos políticos e estratégicos da organização; (c) ajuda a administrar custos e minimizar os riscos técnicos de novas iniciativas; (d) possibilita um equilíbrio constante para um grupo de atividades dentro de um pacote orçamentário pré-estabelecido; (e) oferece informações que podem revelar como projetos aparentemente bem sucedidos podem não gerar retorno suficiente, apoiando assim uma decisão de mudança de foco ou da quantidade de recursos

empregados nesse projeto, antes que muitos recursos sejam consumidos; (f) auxilia na decisão do gerente sobre como diminuir gastos não arbitrários para apoiar novos projetos, onde em muitos casos o orçamento adicional não é garantido; e (g) ajuda a determinar se algum projeto mal sucedido possui valor suficiente, de modo que se indique o desvio de recursos de outro lugar a fim de repará-lo.

De forma similar ao que ocorre no gerenciamento de projetos, as técnicas utilizadas para gerir um portfólio de projetos foram reunidas em um conjunto de processos de gerenciamento do portfólio.

## **2.2. Gerenciamento de Portfólio**

Nos últimos anos, a literatura tem dado grande destaque ao gerenciamento de portfólios e uma série de autores enfatiza a necessidade de se estabelecer processos e procedimentos para o seu gerenciamento de forma eficaz. Segundo RABECHINI JR et al. (2005), o gerenciamento de um número elevado de projetos numa organização, como o cenário que está sendo considerado neste estudo, carece de técnicas que auxiliem os gestores na tomada de decisão quanto ao melhor equacionamento na aplicação de seus recursos. Neste sentido, para que haja melhor coordenação do negócio, é fundamental que haja um esforço para adotar tecnologias que venham incorporar maior disciplina e rigor aos processos da organização quanto à escolha do portfólio.

SPAIR (2009) sugere quatro pontos a serem observados na análise do portfólio da organização: (a) que os projetos selecionados para compor o portfólio estejam bem balanceados, ou seja, correlacionados negativamente, pois dessa forma, caso ocorra algum evento que diminua o valor de algum projeto, outro projeto do portfólio será beneficiado com esse evento, reduzindo assim o risco; (b) que os projetos estejam alinhados com a estratégia da organização, ou seja, o valor de um projeto não pode ser corretamente avaliado sem antes compreender sua prioridade em relação ao plano estratégico da organização; (c) que os projetos do portfólio possam ser concluídos, para que não fiquem pendentes por falta de recursos ou tempo; e (d) que os projetos agreguem valor à organização, ou seja, que o objetivo para o qual o projeto está sendo implementado possa gerar o benefício esperado para a organização.

COOPER et al. (2001) tratam da priorização de projetos e alocação dos recursos disponíveis na organização. Segundo os autores, é de extrema importância haver um

processo de seleção para novos projetos, pois de nada adianta um método muito sofisticado para composição do portfólio da organização se o processo não corta os projetos fadados ao fracasso. Semelhante ao estudo de SPAIR (2009), os autores destacam quatro passos importantes no gerenciamento de portfólio: (a) maximizar o valor do portfólio, selecionando projetos que agreguem mais valor juntos do que agregariam em separado; (b) balancear o portfólio, compensando limitações de um projeto com características de outros; (c) alinhar os projetos que compõem o portfólio com o negócio e a estratégia da organização; e (d) escolher o número certo de projetos para o portfólio, evitando que os projetos dependam de mais recursos do que o disponível e, com isto, fiquem parados até que os recursos estejam disponíveis.

### **2.3. Moderna Teoria do Portfólio**

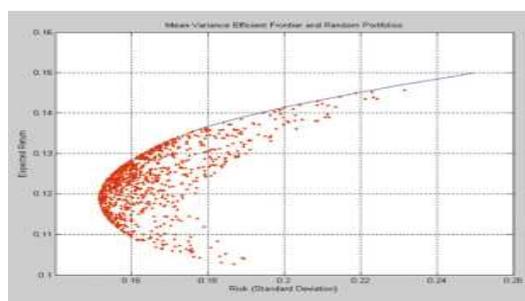
No que trata da seleção de um portfólio de projetos para a organização, atualmente existem muitas técnicas para auxiliar os gerentes nessa tarefa e grande parte delas é baseada na técnica de MARKOWITZ (1952), batizada de Moderna Teoria do Portfólio (MTP).

MTP é uma abordagem originalmente criada para suportar decisões relacionadas a portfólios de investimentos financeiros. A partir de um conjunto de parâmetros sobre o comportamento dos investimentos no passado e expectativas sobre seu comportamento no futuro, a técnica define em quais investimentos e em qual proporção um investidor deve alocar seus recursos de forma a maximizar seu retorno e minimizar os riscos incorridos (JOBSON e KORKIE, 1980).

MARKOWITZ (1952) demonstrou que, dependendo da correlação entre os ativos que compõem um portfólio, é possível reduzir o risco do mesmo a um nível inferior ao de seu ativo mais seguro. Isto é feito combinando-se ativos negativamente correlacionados entre si, de modo a otimizar a relação risco/retorno pela maximização do retorno esperado e minimização da variância.

Dado um conjunto de investimentos candidatos, eles são combinados em diversas proporções para formar um conjunto de portfólios. O retorno esperado e o risco de cada portfólio são calculados com base nos retornos esperados, riscos e dependências entre os investimentos individuais. A representação gráfica dessa solução é um conjunto de diferentes pontos traçados em um plano, onde cada ponto representa um portfólio. Estes

pontos formam uma curva no espaço determinado pelos eixos cartesianos de retorno e risco. Esta curva, chamada de Fronteira Eficiente, representa a camada superior do invólucro de todos possíveis portfólios, indicando aqueles que possuem o menor risco para certo nível de retorno ou o maior retorno para um determinado nível de risco. A Figura 2.1 apresenta um gráfico de fronteira eficiente.



**Figura 2.1 – Gráfico da Fronteira Eficiente**

Fonte: Google Imagens, novembro de 2009.

Um portfólio eficiente é definido como um portfólio que maximiza o retorno esperado para um dado nível de risco ou o portfólio que minimiza o nível de risco para um dado retorno esperado (ENGELS, 2004). Os portfólios eficientes são representados pelos pontos mais próximos da curva formada no gráfico da Fronteira Eficiente.

Sendo  $i$  e  $j$  dois investimentos candidatos a compor o portfólio, o retorno esperado do portfólio ( $RE_p$ ) pode ser representado como:

$$RE_p = w \cdot \mu_i + (1 - w) \cdot \mu_j$$

onde  $w$  é o percentual de recursos investidos em  $i$ ,  $(1 - w)$  é o percentual de recursos investidos em  $j$ ,  $\mu_i$  e  $\mu_j$  são a média de retorno dos investimentos  $i$  e  $j$ , respectivamente.

Por outro lado, o risco do portfólio ( $\sigma_p$ ) é uma função dos riscos dos investimentos individuais que o compõem, do peso de cada investimento no portfólio e das dependências entre os investimentos. Os riscos dos investimentos ( $\sigma_i$  e  $\sigma_j$ ) são representados pelo desvio padrão de seus retornos observados ao longo do tempo. O peso é a proporção dos recursos aplicados em cada investimento e as dependências entre eles são calculadas pela correlação entre seus retornos observados.

A correlação entre dois investimentos  $i$  e  $j$  ( $\rho_{ij}$ ) mostra como um investimento afeta ou é afetado pela presença de outro investimento no mesmo portfólio. A correlação é a representação da força e da direção de um relacionamento entre dois investimentos,

sendo representada por um número entre -1 e +1. Uma correlação igual a -1 indica dois investimentos que se movem em sentido oposto com a mesma intensidade, enquanto uma correlação igual a +1 representa dois investimentos que tendem a se mover na mesma direção e com a mesma intensidade.

O risco de um portfólio composto por dois investimentos  $i$  e  $j$ , onde o investimento  $i$  possui peso  $w$  no portfólio, é calculado pela equação a seguir:

$$\sigma_p^2 = w^2 \cdot \sigma_i^2 + (1 - w)^2 \cdot \sigma_j^2 + 2 \cdot w \cdot (1 - w) \cdot \rho_{ij} \cdot \sigma_i \cdot \sigma_j$$

Dependendo do número de projetos candidatos pode ser impossível avaliar todos os portfólios em um tempo aceitável, utilizando um método exato que se proponha avaliar todos os possíveis cenários e selecionar o melhor dentre eles. Por isso, métodos heurísticos são utilizados com frequência na seleção de portfólios, tanto para portfólios de investimentos quanto para portfólios de projetos.

## 2.4. Técnicas de Otimização para Seleção de um Portfólio

As subseções seguintes apresentam algumas técnicas existentes na literatura sobre otimização da seleção de componentes de portfólios e projetos.

### 2.4.1. Busca Local para Seleção de um Portfólio

SCHAERF (2002) propõe a seguinte formulação para o problema de seleção do portfólio: considere um conjunto de  $n$  investimentos,  $A = \{a_1, \dots, a_n\}$ . Cada investimento  $a_i$  tem um valor de retorno esperado por período  $r_i$  e cada par de investimentos  $\langle a_i | a_j \rangle$  possui uma covariância  $\sigma_{ij}$ . A matriz de covariância  $\sigma_{n \times n}$  é simétrica e cada elemento na diagonal  $\sigma_{ii}$  representa a variância do investimento  $a_i$ . Um portfólio é um conjunto de valores  $X = \{x_1, \dots, x_n\}$ , tal que cada  $x_i$  representa a fração investida em  $a_i$  e um valor positivo  $R$  representa o retorno desejado para o portfólio. O objetivo do problema de seleção de portfólios é minimizar a variância do portfólio assegurando o retorno esperado  $R$ , onde a variância é a medida de risco associada ao portfólio, sendo calculada por  $\sum_{i=1}^n \sum_{j=1}^n \sigma_{ij} \cdot x_i \cdot x_j$ .

O autor tinha como objetivo investigar, comparar e combinar diferentes estratégias de busca local para resolver uma versão do problema da seleção de portfólios com um conjunto de restrições. As restrições impostas são:

- (i) Restrição de cardinalidade, onde o número de investimentos que compõem o portfólio é limitado. Isto é, um valor  $k \leq n$  é dado, tal que o número de  $i$ 's para o qual  $x_i > 0$  é no máximo  $k$ ; e
- (ii) Restrição de quantidade, onde a quantidade de cada investimento  $i$  incluída no portfólio é limitada a um dado intervalo. Especificamente, é dado um mínimo  $\varepsilon_i$  e um máximo  $\delta_i$  para cada investimento  $i$ .

A busca local consiste em definir o espaço de busca e uma relação de vizinhança entre as possíveis soluções  $S_i$ , criando um estado inicial  $s_0$  e movendo de  $s_0$  até  $s_{\text{final}}$  de acordo com uma estratégia escolhida. Para que fossem aplicadas técnicas de busca local (*Hill Climbing, Simulated Annealing e Tabu Search*) para a seleção de portfólios, o autor definiu o espaço de busca, a estrutura da vizinhança e uma função de custo. Para representar o espaço de busca, foram utilizadas duas seqüências,  $L = \{a_1, \dots, a_p\}$  e  $S = \{x_1, \dots, x_p\}$ , tal que  $x_i$  é a fração de  $a_i$  no portfólio, ou seja, a quantidade que esse investimento representa no portfólio. O conjunto  $L$  representa um subconjunto de  $A$ , enquanto  $S$  representa um subconjunto de  $X$ . Todos os projetos  $a_j$  que pertencem a  $A$  e não pertencem a  $L$  tem a fração  $x_j$  implicitamente igual a zero.

Foram estabelecidas três relações de vizinhança para satisfazer a restrição de cardinalidade:

- IDR ([I]ncrementar, [D]ecrementar, t[R]ocar): a quantidade de um investimento escolhido é aumentada ou diminuída no portfólio. Todas as partes dos outros investimentos são alteradas, de forma que o portfólio continue viável. Se a parte do investimento escolhido ficar abaixo do mínimo, ele deve ser substituído por outro investimento;
- IDID ([I]ncrementar, [D]ecrementar, [I]nserir, [D]eletar): é similar ao IDR, exceto que o investimento excluído não é substituído. Também se considera aqui a inclusão de novos investimentos;
- TID ([T]ransferir, [I]nserir, [D]eletar): uma parte do recurso é transferida de um investimento a outro. A transferência pode ser feita também a um investimento que não esteja no portfólio, que é então inserido. Se a parte do investimento cair abaixo do mínimo, este deve ser excluído do portfólio.

Para a função de custo, foram definidos dois componentes:  $f_1(X) = \max(0, \sum_{i=1}^n r_i x_i - R)$  e  $f_2(X) = \sum_{i=1}^n \sum_{j=1}^n \sigma_{ij} x_i x_j$ , que levam em consideração o grau de violação da restrição de cardinalidade e a função objetivo, respectivamente. A função de custo total das funções  $f_1(X)$  e  $f_2(X)$  é a combinação linear:  $f(X) = w_1 f_1(X) + w_2 f_2(X)$ , onde  $w_1$  e  $w_2$  representam o peso. Durante a busca, o peso  $w_1$  é levado a variar de acordo com um mecanismo de penalidade para o movimento: se a restrição de cardinalidade for satisfeita para  $K$  iterações consecutivas,  $w_1$  é dividido por um fator randômico, escolhido entre 1.5 e 2. Do contrário, se a restrição de cardinalidade for violada para  $H$  iterações consecutivas,  $w_1$  é multiplicado por um fator randômico de mesma escala, sendo  $H$  e  $K$  parâmetros do algoritmo.

O autor explorou o uso de técnicas de busca local, comparando e combinando trabalhos anteriores de seleção de portfólios que fazem uso dessa abordagem e propôs novos algoritmos que combinam diferentes relações de vizinhança. Chegou à conclusão de que a técnica de *Tabu Search* foi a que teve melhores resultados e contestou o trabalho de CHANG et al. (2000), que sugeriu que a técnica de *Simulated Annealing* apresentava melhores resultados quando comparada com a técnica de *Tabu Search*. Os resultados obtidos por SCHAERF (2002) podem ser consultados através do endereço eletrônico <http://mscmga.ms.ai.ac.uk/jeb/orlib/portfolio.html>.

Apesar de a técnica apresentar bons resultados para portfólios de investimentos financeiros, como compra e venda de ações, sua utilização em projetos de TI pode não ser indicada, pois a formulação da solução desenvolvida ao longo da pesquisa de SCHAERF (2002) permite que a proporção do capital aplicado em cada investimento financeiro seja determinado pelo investidor e que esta proporção possa ser alterada a qualquer momento. No entanto, em projetos de desenvolvimento de software, a proporção do capital investido é definida como sendo a quantia necessária para desenvolver o projeto e muitas vezes não é permitida a alteração dessa quantia.

#### **2.4.2. Busca Heurística para Seleção de um Portfólio**

O trabalho de BAKAR et al. (2009) apresenta os resultados de uma técnica de otimização de portfólios que utiliza algoritmos genéticos e os compara com o método convencional de MARKOWITZ (1952). Para isso, foi selecionado um conjunto de dez investimentos dentre os dois principais setores de Bursa Malaysia, uma das maiores

Bolsas da Ásia. A partir desses investimentos foram montados 25 portfólios para análise e comparação das técnicas na busca pelo melhor índice de retorno dos portfólios.

A técnica utiliza um algoritmo genético que identifica cada cromossomo, ou solução candidata, como um vetor que contém o retorno esperado para o portfólio, o risco associado a ele e o percentual de tolerância do investidor acerca do risco. Inclui no processo as operações de cruzamento, mutação e seleção. No cruzamento, dois portfólios P1 e P2 geram filhos P3 e P4 com base em um número aleatório selecionado dentro de uma faixa definida pelo gestor. Dependendo deste valor, os portfólios gerados podem receber investimentos provenientes de P1 ou de P2. Na mutação, uma variável do cromossomo é trocada por um valor aleatório entre o retorno esperado do portfólio e sua variância. Na seleção, há a separação da metade dos indivíduos mais aptos para reprodução, ou seja, todos os indivíduos que possuam o valor da função objetivo maior do que a média geral.

Vários testes foram executados utilizando diferentes níveis de tolerância ao risco e foi observado que as duas técnicas mostraram similaridade ao apontar o portfólio com maior valor utilidade, demonstrando que algoritmos genéticos podem ser utilizados para a resolução do problema da seleção de portfólios.

#### **2.4.3. Busca Heurística para Seleção de um Portfólio em Estágios**

LAI et al. (2006) propuseram o uso de um algoritmo genético para cada estágio de uma abordagem para otimização de um portfólio de investimentos financeiros. Dado um conjunto de investimentos, um algoritmo genético é proposto para identificar aqueles de boa qualidade, ou seja, os que possuem maiores taxas de retorno com riscos menores, para então priorizá-los. Em um segundo estágio, é feita uma otimização para o problema da alocação dos recursos entre os investimentos selecionados, utilizando um algoritmo genético baseado na teoria de MARKOWITZ (1952). Através destas duas fases, um portfólio ótimo pode ser determinado.

Para determinar a qualidade de cada investimento são utilizados quatro indicadores financeiros disponibilizados pelas próprias organizações em seus relatórios anuais. Estes indicadores e suas fórmulas de cálculo são apresentados a seguir.

**Tabela 2.1 - Representação dos indicadores usados por LAI et al. (2006)**

Retorno do Capital Empregado (RDCE)	$RDCE = (\text{Lucro}) / (\text{Capital Líquido dos Acionistas}) * 100\%$
Ganho por Investimento (GPI)	$GPI = (\text{Rendimento Líquido}) / (\text{Número de Ações Ordinárias})$
Relação Custo/Lucro (Relação C/L)	$\text{Relação C/L} = (\text{Preço do Investimento}) / (\text{Ganho por Ação}) * 100\%$
Liquidez	$\text{Liquidez} = (\text{Bens Correntes}) / (\text{Dívidas Correntes}) * 100\%$

Uma vez obtidos os valores destes indicadores para cada um dos investimentos, um algoritmo genético é utilizado para identificar a qualidade desses investimentos. Os passos do algoritmo genético são: (a) a criação da população inicial; (b) avaliação do cromossomo com uma função objetivo; e (c) aplicação dos operadores de seleção, cruzamento e mutação. A população inicial é gerada com uma codificação binária da informação obtida com os quatro indicadores financeiros, cujo padrão varia de 0 a 7, onde zero representa valores baixos do indicador financeiro, ou seja, um investimento deficiente nesta análise, e sete representa a obtenção dos valores mais altos. Quatro genes representando a codificação dos quatro indicadores financeiros formam um cromossomo, que por sua vez representa um investimento. A figura 2.2 mostra a representação de um investimento.



**Figura 2.2 – Esquema de representação de um investimento**

Uma vez que os valores estimados dos quatro indicadores estejam calculados e atribuídos aos quatro respectivos genes do cromossomo, é feita a soma desses valores para cada cromossomo e então priorizados os cromossomos com maior valor obtido nessa soma, gerando uma classificação  $R_{derivado}$ .

O próximo passo é avaliar todos os cromossomos gerados através de uma função objetivo, que no estudo foi utilizada para minimizar a raiz do erro médio quadrático. No entanto, a raiz do erro médio quadrático é uma métrica utilizada para estimar a eficiência e qualidade de um classificador e serve para medir a diferença entre os valores previstos por um modelo e os valores reais observados do objeto que está sendo modelado ou estimado. Como o resultado gerado até agora é uma classificação estimada

dos cromossomos, ou seja, a classificação derivada dos indicadores financeiros ( $R_{derivado}$ ), os autores adicionam o uso de valores reais observados para gerar outra classificação dos investimentos ( $R_{real}$ ), medida pelo Retorno do Custo Anual (RCA), calculado pela fórmula a seguir:

$$RCA_n = \frac{\text{Preço do Investimento}_n - \text{Preço do Investimento}_{n-1}}{\text{Preço do Investimento}_{n-1}}$$

onde  $RCA_n$  é o retorno do custo anual para o ano  $n$  e  $\text{Preço do Investimento}_n$  é o custo do projeto para o ano  $n$ , sendo estes valores informados pelo gestor do projeto, que coletará estas informações de algum banco de dados disponível na organização. Usualmente, os investimentos com um alto valor RCA são considerados bons investimentos.

Com o valor do RCA avaliado para cada um dos  $N$  investimentos, eles serão associados a uma classificação  $r$ , variando de 1 a  $N$ , onde o primeiro elemento representa o mais alto valor de RCA, enquanto que o  $n$ ésimo representa o mais baixo. Por conveniência, para a comparação dos resultados entre o  $R_{derivado}$  e o  $R_{real}$ , a classificação  $r$  será mapeada linearmente no intervalo de 0 a 7, de acordo com a equação abaixo:

$$R_{real} = 7 \cdot \frac{N - r}{N - 1}$$

Na sequência, os genes que representam os indivíduos são avaliados pela função objetivo, que busca minimizar a raiz do erro médio quadrático (ou REMQ) na diferença entre a classificação derivada dos indicadores financeiros ( $R_{derivado}$ ) e a classificação real obtida através do histórico observável ( $R_{real}$ ), para um cromossomo particular. Abaixo está a representação da equação:

$$REMQ = \sqrt{\frac{1}{N} \sum_{t=1}^N (R_{derivado} - R_{real})^2}$$

onde  $R_{derivado}$  representa os cromossomos classificados pelo valor recebido através dos quatro indicadores financeiros,  $R_{real}$  representa os cromossomos classificados pelo Retorno do Custo Anual,  $N$  é o número de cromossomos e  $t$  representa o tempo na medição dos valores.

Após desenvolver a população, os cromossomos com maior aptidão são selecionados através do algoritmo da roleta (*Roulette Wheel Method*) (MICHALEWICZ, 1996). Este

método aloca para cada cromossomo um espaço proporcional ao seu valor aptidão, ou seja, os mais aptos possuem mais chances de serem selecionados. No passo seguinte, um operador de cruzamento de um corte (*one-point crossover*) (MOTA, 1996) é aplicado para gerar novos cromossomos descendentes, onde dois genes dentro de um mesmo cromossomo sofrem intercâmbio em um ponto de cruzamento selecionado aleatoriamente pelo referido operador, dando origem a um cromossomo filho. Depois disso, os outros cromossomos ficam com uma probabilidade de 0.005 de sofrerem mutação, que altera um de seus genes randomicamente de zero para um ou vice versa. A mutação evita que o algoritmo convirja muito rapidamente para um mínimo local.

Após a seleção dos melhores investimentos, a segunda parte do trabalho leva em consideração o risco dos investimentos, conforme a teoria do portfólio de MARKOWITZ (1952). Os valores de retorno esperado e covariância são calculados para todos os investimentos para que seja estabelecida a quantidade de recursos que devem ser empregados em cada investimento e para minimizar o risco, dado certo nível de retorno esperado. As formulações podem ser vistas abaixo:

Retorno Esperado

$$E(R_i) = \sum_{i=1}^n R_{it} / n$$

onde:

$$R_{it} = \frac{SCP_{it} - SCP_{i(t-1)}}{SCP_{i(t-1)}}$$

Covariância

$$\sigma_{i,t} = \frac{1}{n} \sum_{i=1}^n (R_{i,t} - E(R_i)) \cdot (R_{j,t} - E(R_j))$$

sendo  $R_{it}$  o retorno do investimento  $i$  no tempo  $t$ ,  $SCP_{it}$  o preço do fechamento no tempo  $t$  para o investimento  $i$  e  $n$  o número de períodos de tempo para os dados avaliados.

Na função objetivo desta parte da abordagem, os autores buscam minimizar o risco controlando a proporção dos investimentos no portfólio e tentando alcançar o retorno pré-determinado pelo gestor. A formulação da função objetivo é a combinação desses dois objetivos e está descrita abaixo:

$$\sum_{i=1}^n \sum_{j=1}^n \sigma_{ij} x_i x_j + (\sum_{i=1}^n E(R_i) x_i - R_p^*)^2$$

onde  $\sigma_{ij}$  é a covariância dos investimentos  $i$  e  $j$ ,  $x_i$  é a proporção do investimento  $i$  no portfólio e  $R_p^*$  é o retorno pré-determinado pelo gestor. Através destes dois estágios de

otimização, a técnica busca os portfólios que possuam maior probabilidade de retorno e combinação ótima de recursos aplicados em cada investimento. Entretanto, a técnica é utilizada apenas para investimentos financeiros, não podendo ser totalmente aplicada para projetos de software, pois depende em sua formulação do histórico observável dos projetos de software. Projetos são únicos por definição (PMI, 2008) e não possuem dados históricos para serem coletados.

#### 2.4.4. Busca Heurística para Seleção de um Portfólio com Restrição de Lote

No estudo de LIN e LIU (2008) são propostos três modelos de decisão para seleção de um portfólio de investimentos com restrição de lote mínimo para cada negociação efetuada, bem como algoritmos genéticos para obter as soluções. Segundo a BM&FBOVESPA, toda organização tem suas ações negociadas em lotes, que podem ser de uma ação, dez ações, 100 ações, etc. Essa quantidade de ações representa o lote mínimo que pode ser negociado no mercado padrão.

Na proposta de LIN e LIU (2008) o primeiro modelo para seleção de um portfólio de investimentos foi construído a partir do modelo de MARKOWITZ (1952). O segundo modelo utiliza a formulação do primeiro modelo, mas minimiza a distância entre o portfólio esperado e o portfólio obtido. O terceiro modelo é derivado de uma abordagem *fuzzy* multi-objetivo para alcançar o retorno e o risco desejado. A diferença primária entre esses três modelos reside em suas funções objetivo, ocasionando diferentes funções aptidão para cada um dos algoritmos genéticos.

Existem outros métodos baseados em modelos lineares de otimização de carteiras que consideram lotes mínimos de transação (KONNO e WIJAYANAYAKE, 2001, MANSINI e SPERANZA, 1997, MANSINI e SPERANZA, 1999, MANSINI e SPERANZA, 2005), mas, segundo os autores, nenhum deles é baseado no modelo de Markowitz. Para considerar o lote mínimo de transação, os autores modificaram o modelo original de Markowitz, conforme mostrado a seguir:

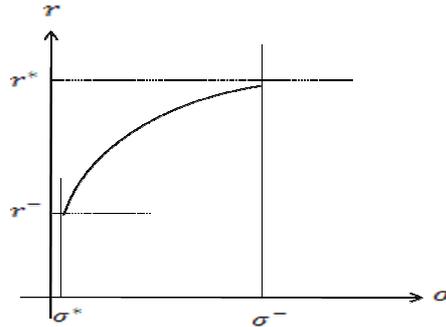
$$\begin{aligned} \text{Min} \quad & \sigma_p^2 = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \sigma_{ij} \\ \text{Sujeito a} \quad & e_p = \sum_{i=1}^n x_i c_i r_i \geq br \\ & \sum_{i=1}^n x_i c_i \leq b \\ & w_i = \frac{c_i x_i}{\sum_{i=1}^n c_i x_i}, i = 1, \dots, n \end{aligned}$$

onde  $n$  é o número de investimentos candidatos,  $r_i$  é a taxa de retorno esperada do investimento  $i$ ,  $\sigma_{ij}$  é a covariância entre os investimentos  $i$  e  $j$ ,  $\sigma_p^2$  é a variância do portfólio,  $b$  é o orçamento disponível para investimento,  $w_i$  é a proporção do orçamento destinado ao investimento  $i$ ,  $r$  é a taxa do retorno desejada,  $e_p$  é o retorno esperado do portfólio,  $c_i$  é a unidade de preço do investimento  $i$  e  $x_i$  representa a proporção do investimento  $i$  no portfólio. O modelo mostrado acima minimiza o risco do investimento para dado retorno  $r$  e possui solução inteira que corresponderá às unidades negociáveis dos investimentos.

Segundo os autores, o risco de um portfólio tende a se elevar quando o seu retorno aumenta. Com isso, é necessário verificar se a taxa  $e_p$  de retorno excede a taxa  $r$  do retorno desejado, o que pode fazer com que o risco da solução ótima deste modelo reformulado seja muito alto para ser aceitável. Assim, um portfólio com uma taxa de retorno ligeiramente menor que o  $r$  desejado pode ser mais aceitável se o risco do portfólio for reduzido de maneira significativa. Esta possibilidade levou os autores a outro modelo de decisão, que minimiza a distância entre o portfólio obtido e o portfólio desejado, através da formulação abaixo:

$$\begin{aligned} \text{Min} \quad & d^2 = \left( \frac{\sigma_p - \sigma}{\sigma^- - \sigma^*} \right)^2 + \left( \frac{e_p - br}{br^* - br^-} \right)^2 \\ \text{Sujeito a} \quad & \sigma_p^2 = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \sigma_{ij} \\ & e_p = \sum_{i=1}^n x_i c_i r_i \\ & \sum_{i=1}^n x_i c_i \leq b \\ & w_i = \frac{c_i x_i}{\sum_{i=1}^n c_i x_i}, \quad i = 1, \dots, n \end{aligned}$$

onde  $\sigma$  é o menor risco correspondente ao  $r$  desejado,  $r^*$  representa a taxa máxima de retorno correspondente ao menor nível de risco que pode ser alcançado,  $\sigma^*$  representa o menor nível de risco que pode ser alcançado,  $r^-$  e  $\sigma^-$  representam seus correspondentes opostos, ou seja,  $(r^*, \sigma^-)$  e  $(r^-, \sigma^*)$  estão nas extremidades da fronteira eficiente, conforme mostrado na figura 2.3.



**Figura 2.3 – Gráfico representativo da fronteira eficiente**

Fonte: Google Imagens, novembro de 2009.

A vantagem deste modelo sobre o anterior é que ele permite que a taxa de retorno do portfólio seja menor que a taxa de retorno desejada. Mas apesar de fornecer boas soluções para o problema proposto, o modelo ainda possui um problema: uma mesma taxa de retorno desejada pode envolver diferentes níveis de risco dentre as soluções candidatas, não sendo possível prever um valor apropriado para  $r$  de modo que se reflita o nível de aversão ao risco do acionista. Por este motivo, foi desenvolvido outro modelo baseado na programação *fuzzy*, em que é possível expressar o nível de aversão ao risco do investidor e informar ainda o retorno desejado através de parâmetros dentro do intervalo  $[0,1]$ .

A programação *fuzzy* tem sido amplamente utilizada na resolução de problemas de tomada de decisão (LIN, 2004). A formulação geral é apresentada abaixo:

$$\begin{array}{ll} \text{Encontrar} & x, \\ \text{Para satisfazer} & f_j(x) \geq \sim g_j, j = 1, 2, \dots, m, \\ \text{Sujeito a} & Bx \leq b, x \geq 0, \end{array}$$

onde  $x$  é a solução no vetor formado pelos componentes  $x_1, x_2, \dots, x_n$ ,  $f_j$  representa o objetivo de número  $j$  e  $Bx \leq b$  são as restrições impostas ao vetor. A notação  $\geq \sim$  significa que a relação é fuzzy, tal que o objetivo  $g_j$  pode ser parcialmente alcançado.

Na criação da função de pertinência, os autores utilizaram o trabalho de LIN (2004), que propôs um modelo Max-Min com atribuição de peso para problemas com múltiplos objetivos. O modelo pode ser representado por:

$$\begin{array}{ll} \text{Max} & \lambda, \\ \text{Sujeito a} & \omega_j \lambda \leq \frac{f_j(x) - l_j}{g_j - l_j}, j = 1, 2, \dots, m, \\ & Bx \leq b, x \geq 0 \end{array}$$

onde  $\lambda$  representa o nível mínimo alcançado entre os objetivos,  $\omega_j$  é o peso do objetivo de número  $j$ ,  $f_j(x)$  é o objetivo de número  $j$ ,  $g_j$  representa um determinado objetivo e  $l_j$  é o menor limite de tolerância para o objetivo de número  $j$ . Com o modelo acima, os autores utilizaram como objetivos o retorno e o risco. Então, sejam  $br^*$  e  $\sigma^*$  os objetivos e  $br^-$  e  $\sigma^-$  os limites de tolerância. As funções de pertinência para o retorno e risco são:

$$\mu_r(e_p) = \begin{cases} 1 & \text{Se } br^* \leq e_p \\ \frac{e_p - br^-}{br^* - br^-} & \text{Se } br^- \leq e_p < br^* \\ 0 & \text{Se } e_p < br^- \end{cases} \quad \mu_\sigma(\sigma_p) = \begin{cases} 1 & \text{Se } \sigma_p \leq \sigma^* \\ \frac{\sigma^- - \sigma_p}{\sigma^- - \sigma^*} & \text{Se } \sigma^* \leq \sigma_p < \sigma^- \\ 0 & \text{Se } \sigma^- < \sigma_p \end{cases}$$

Sendo  $\omega_r$  e  $\omega_\sigma$  os pesos para o retorno e risco, foi formulado o seguinte modelo fuzzy:

$$\begin{aligned} \text{Max} & \quad \lambda, \\ \text{Sujeito a} & \quad \omega_r \lambda \leq \frac{e_p - br^-}{br^* - br^-}, \\ & \quad \omega_\sigma \lambda \leq \frac{\sigma^- - \sigma_p}{\sigma^- - \sigma^*}, \\ & \quad \sigma_p^2 = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \sigma_{ij} \\ & \quad e_p = \sum_{i=1}^n x_i c_i r_i \\ & \quad \sum_{i=1}^n x_i c_i \leq b \\ & \quad w_i = \frac{c_i x_i}{\sum_{i=1}^n c_i x_i}, \quad i = 1, \dots, n \end{aligned}$$

Com o modelo acima, a preferência do gestor poderá ser informada. Por exemplo, acionistas com grande aversão a riscos podem dar maior peso ao risco  $\omega_\sigma$  do que ao retorno  $\omega_r$ , contanto que  $\omega_\sigma + \omega_r = 1$ .

Para a construção dos algoritmos genéticos, diversas estratégias poderiam ser utilizadas na busca por soluções para os três modelos propostos. Dentre elas: (i) rejeição, que procura excluir as soluções inviáveis imediatamente na geração; (ii) reparação com estratégias de penalidade, que transforma uma solução ruim em uma solução viável, através de algum processo de reparação (MA et al., 2006); e (iii) modificação de operadores genéticos, que busca planejar representações específicas do problema com operadores genéticos especializados para manter a viabilidade da solução (GEN e CHENG, 1997).

Os autores optaram por utilizar estratégias de penalidade e modificação dos operadores genéticos, bem como utilizar números reais na codificação do peso  $w_i$  ao invés de codificar diretamente as unidades negociadas  $x_i$  como inteiros. Assim, cada gene representa o peso do orçamento destinado ao investimento  $i$  e será descrito por um

número real de 0 a 1. No entanto, no estágio inicial ou no estágio após as operações genéticas, a soma desses pesos pode não chegar ao total de 100% e por isso os pesos devem ser normalizados. Com os pesos normalizados, as unidades negociadas do investimento  $i$  podem ser obtidas pela equação:

$$x_i = \left\lfloor \frac{bw'_i}{c_i} \right\rfloor, i = 1, \dots, n, \text{ onde } w' \text{ é o novo valor do peso } w \text{ após a normalização.}$$

Por fim, três algoritmos genéticos foram desenvolvidos para testar os três modelos propostos, cada qual com sua função objetivo.

Para o primeiro modelo, os autores utilizam uma função de penalidade com o intuito de satisfazer a restrição  $e_p = \sum_{i=1}^n x_i c_i r_i \geq br$ .

$$P(x) = \begin{cases} br - \sum_{i=1}^n x_i c_i r_i & \text{Se } br > \sum_{i=1}^n x_i c_i r_i \\ 0 & \text{Caso contrário.} \end{cases}$$

Incorporando a função objetivo descrita no primeiro modelo, que trata da minimização do risco do portfólio, com a função penalidade apresentada acima, a função objetivo para o primeiro modelo é:

$$\text{Aptidão} = \exp(-k (\sigma_p^2 + M \cdot P(x)))$$

onde  $k$  é uma constante positiva,  $M$  é um número positivo muito grande que serve para forçar uma solução com retorno maior ou igual ao estabelecido na restrição e a função exponencial com a constante  $k$  estabelece o limite da avaliação e assim alivia a pressão da seleção por cromossomos com alto valor de aptidão, prevenindo que o algoritmo convirja muito rapidamente.

O segundo modelo é também um problema de minimização do risco, tal como no primeiro modelo, mas leva em consideração o  $d^2$ , que busca diminuir a distância entre o portfólio obtido e o portfólio desejado. Isso é feito para evitar que o retorno seja muito alto, o que elevaria muito o nível do risco. Assim, a função objetivo para este problema é definida de maneira similar:

$$\text{Aptidão} = \exp(-kd^2)$$

A função objetivo para o terceiro modelo, que é um problema de maximização, é simplesmente sua função objetivo  $\lambda$ :

$$\text{Aptidão} = \min \left\{ \frac{e_p - br^-}{\omega_r(br^* - br^-)}, \frac{\sigma^- - \sigma_p}{\omega_\sigma(\sigma^- - \sigma^*)} \right\}$$

Para validar os três modelos, os autores fizeram um estudo empírico utilizando dados da bolsa de Taiwan entre os anos de 1997 até 2000. De acordo com esses estudos, os três modelos obtiveram, em poucos minutos, soluções próximas do esperado, utilizando em média 96% do orçamento. Este é também um modelo voltado para portfólios de investimentos financeiros, que utiliza a abordagem de lote mínimo de transação, conceito que não é aplicado em projetos de desenvolvimento de softwares.

#### **2.4.5. Busca Heurística de um Portfólio usando Diferentes Medidas de Risco**

CHANG et al. (2009) propuseram uma abordagem heurística baseada em algoritmos genéticos para o problema da otimização de portfólios utilizando diferentes medidas de risco. O intuito é demonstrar que o problema da otimização de portfólios pode ser resolvido com sucesso através de algoritmos genéticos, mesmo que o nível de risco do portfólio seja medido através de métricas diferentes. As métricas utilizadas para medir o nível de risco foram: (i) semi-variância; (ii) desvio médio absoluto; e (iii) variância com valor de assimetria.

O algoritmo genético utilizado por CHANG et al. (2009) gera, aleatoriamente, uma população inicial com um número constante de cromossomos. Cada cromossomo representa o peso de cada investimento individual no portfólio, sendo estes pesos alterados ao longo do processo de otimização para alcançar a composição ideal de investimentos. Uma função objetivo é usada para avaliar a aptidão de cada cromossomo, definindo quão boa é a solução representada por ele. Em seguida, são aplicados os operadores genéticos de cruzamento e mutação.

No cruzamento, cada par de indivíduos gera um descendente que possuirá determinado investimento  $i$  se ambos os indivíduos pais possuírem esse investimento. Se apenas um dos pais tiver o investimento  $i$ , a probabilidade de que o descendente o possua será de 50%. Na mutação, a proporção do indivíduo no portfólio é aumentada ou diminuída em 10% do valor da proporção de um investimento  $i$  qualquer, selecionado de forma aleatória, atentando para o limite máximo que cada indivíduo pode ocupar.

De acordo com os resultados do trabalho, vemos que o modelo do algoritmo proposto é eficiente para resolver o problema da otimização do portfólio: utilizando diferentes medidas de risco, os acionistas podem encontrar a fronteira eficiente baseados em uma quantidade fixa de investimentos.

## 2.5 Conclusão

Nos diversos trabalhos estudados, foi possível ver a aplicação de técnicas e métodos heurísticos para ajudar os gestores na tomada de decisão sobre a seleção de projetos ou investimentos para a composição de seus portfólios. A ampla utilização de algoritmos genéticos nos métodos e técnicas propostas indica a boa aceitação dessa abordagem na comunidade científica e que o uso da Moderna Teoria do Portfólio proporciona informações importantes para que o gestor faça boas escolhas acerca dos projetos para seu portfólio. Entretanto, técnicas que possam ser utilizadas para selecionar e priorizar não apenas investimentos financeiros para o portfólio ainda são muito raras e pouco exploradas, tornando esta área de estudo relevante.

No próximo capítulo, apresentaremos um modelo formal para o problema de seleção de portfólios de projetos e o funcionamento da técnica proposta por COSTA (2011), bem como sua limitação quanto ao número de projetos candidatos a fazer parte do portfólio, para então apresentarmos nossa proposta de solução para o tratamento dessa limitação.

# PROPOSTA DE SOLUÇÃO PARA A TÉCNICA DE SELEÇÃO DE PORTFÓLIO DE PROJETOS

---

### 3.1. Considerações Iniciais

A alta complexidade dos projetos de software da atualidade demanda investimentos de tempo, trabalho, dinheiro e outros recursos que, se não forem bem empregados, refletirão em desperdício ou mesmo no fracasso de projetos. Por isso, o portfólio de projetos de uma organização precisa ser bem montado, selecionando apenas os projetos que possuam maior probabilidade de alcançar seus objetivos, maximizando os lucros e minimizando os riscos da organização que administra o portfólio.

Para selecionar ou priorizar projetos para o portfólio de uma organização, os gestores costumam adotar técnicas que os auxiliem nesta tomada de decisão. Existem diversas técnicas de balanceamento de portfólio na literatura, tal como as descritas no Capítulo 2. Neste Capítulo será descrito o funcionamento da técnica de apoio à seleção de portfólio de projetos de software apresentada em (COSTA, 2011), cuja fundamentação é feita na Moderna Teoria do Portfólio, proposta por (MARKOWITZ, 1952).

O objeto de estudo desta dissertação é uma limitação da técnica, que a impede de ser utilizada em cenários com grande número de projetos candidatos a participar do portfólio. Isto ocorre porque a técnica analisa todas as possíveis combinações de projetos com o intuito de selecionar o portfólio mais adequado sob o aspecto da relação risco/retorno e que possa ser executado com um volume previamente estabelecido de recursos. À medida que aumenta o número de projetos candidatos a compor o portfólio, o número de combinações a serem examinadas cresce exponencialmente, impossibilitando o exame de todas as alternativas em tempo aceitável.

A proposta deste trabalho é utilizar uma heurística baseada em algoritmos genéticos para criar uma implementação da técnica supracitada, viabilizando o seu uso em situações em que o universo de projetos candidatos não permite a análise completa de todas as suas combinações. Esta estratégia não garante que o melhor portfólio de projetos seja selecionado, mas pode oferecer um portfólio de qualidade (segundo a relação de risco e retorno computados para cada portfólio) em tempo menor do que o necessário para examinar todas as possíveis combinações de projetos candidatos.

Este Capítulo está dividido em seis seções. A Seção 3.1 apresenta esta introdução. A Seção 3.2 apresenta conceitos relacionados a buscas heurísticas. A Seção 3.3 apresenta a técnica de apoio à seleção de portfólio de projetos de software proposta por (COSTA, 2011), as entidades envolvidas e seus relacionamentos. A Seção 3.4 exhibe a formalização do modelo proposto. A Seção 3.5 descreve a proposta de solução para o problema e, finalmente, a Seção 3.6 encerra o Capítulo.

### **3.2. Busca Heurística com Algoritmos Genéticos**

Problemas intratáveis ou difíceis são comuns na natureza e nas áreas de conhecimento. Atualmente, existem diversos problemas que podem ser considerados como intratáveis computacionalmente (GOLDBERG, 2000) e dentre todos esses problemas destacam-se os problemas de otimização combinatória, cuja finalidade é buscar soluções para maximizar ou minimizar determinada função objetivo ou função custo, atendendo a um conjunto de restrições (PAPADIMITRIOU e STEIGLITZ, 1998). Esses problemas são considerados intratáveis quando não podem ser resolvidos com algoritmos determinísticos em tempo polinomial (TOSCANI e VELOSO, 2002). Isso significa que resolver um problema intratável pelo método tradicional de busca exaustiva (força bruta), onde todas as possíveis soluções do espaço de busca são analisadas, acarreta um tempo de computação que, para entradas suficientemente grandes, levaria séculos para gerar o resultado (PAPADIMITRIOU e STEIGLITZ, 1998).

Uma alternativa para se buscar uma solução viável para estes problemas é a utilização de métodos heurísticos, que consistem de algoritmos que buscam uma solução aproximada para o problema, uma vez que muitos problemas não requerem uma solução exata (TOSCANI e VELOSO, 2002).

Heurísticas são técnicas que buscam boas soluções para problemas combinatórios sem fazer a análise de todas as alternativas possíveis, ou seja, procuram uma solução que esteja próxima da solução ótima dentro do espaço de soluções candidatas para o problema (REEVES, 1993). Contudo, as heurísticas não oferecem garantias na qualidade e viabilidade das soluções geradas. Ainda assim, são técnicas frequentemente adotadas por apresentarem baixo custo computacional e por criarem soluções de qualidade muito superior à média das soluções aleatórias (NETTO, 2010).

No problema da seleção de portfólio de projetos, dependendo do número de projetos candidatos e da avaliação de seus riscos, pode ser impossível encontrar, em tempo aceitável, uma solução ótima com um método exato que se proponha a avaliar todas as possíveis alternativas de solução. Por isso, é apresentada nesta dissertação a proposta de utilizar um método heurístico para abordar este problema. Entre os possíveis métodos heurísticos que poderiam ser aplicados, a Computação Evolucionária, mais especificamente a técnica de algoritmos genéticos, tem sido usada com sucesso para solucionar esse tipo de problema, como pode ser visto no Capítulo 2 em (BAKAR et al., 2009), (CHANG et al., 2009), (LIN e LIU, 2008) e (LAI et al., 2006).

Algoritmos genéticos propõem a implementação computacional dos conceitos elementares da genética para solução de problemas de otimização (HOLLAND, 1962). Utilizando o princípio de evolução genética da biologia, a Computação Evolucionária aplica operações de seleção, cruzamento e mutação em uma população na qual cada indivíduo representa uma solução no espaço de busca das soluções candidatas de um determinado problema.

Segundo GOLDBERG (2009), os algoritmos genéticos diferem dos métodos tradicionais de busca e otimização por: (i) trabalharem com a codificação de um conjunto de parâmetros e não com os próprios parâmetros; (ii) fazerem busca sobre uma população de pontos e não sobre um único ponto; (iii) utilizarem funções de custo (funções objetivo) e não derivadas ou outro tipo de conhecimento auxiliar; e (iv) por utilizarem regras de transição probabilísticas e não determinísticas.

Os principais componentes de um algoritmo genético são: (i) indivíduo ou cromossomo, que representa uma solução para um dado problema; (ii) gene, que representa uma característica ou parte da solução; (iii) população, que representa o conjunto de indivíduos ou soluções de uma mesma geração; função objetivo (*fitness*),

que é um mecanismo de estimativa da qualidade dos indivíduos ou soluções geradas, ou seja, uma função que atribui um valor de avaliação para cada indivíduo ou solução do problema; (iv) pais (*parents*), que são os indivíduos selecionados como aptos em uma geração para transmitirem suas características para uma nova prole ou indivíduos filhos (*offspring*); (v) filhos (*offspring*), que representam a nova prole ou os novos indivíduos, gerados a partir de operações realizadas sobre os indivíduos pais; (vi) mecanismo de cruzamento (*crossover*), que opera sobre os indivíduos fazendo a combinação de partes (genes) de dois ou mais indivíduos de uma dada população; (vii) mecanismo de mutação, que opera sobre os genes dos indivíduos da população, de acordo com uma taxa ou probabilidade; e (viii) mecanismo de seleção, que faz uma seleção de indivíduos orientada aos objetivos do problema para evoluir uma solução ideal.

A estrutura geral de um algoritmo genético pode ser definida como: (i) pegar uma população de soluções candidatas para resolver um determinado problema; (ii) aplicar operadores inspirados nos mecanismos de evolução genética natural sobre esta população; (iii) aplicar seleção orientada aos objetivos do problema para evoluir uma solução ideal; e (iv) gerar nova população de soluções até que algum determinado critério de parada seja alcançado.

### **3.3. O Problema de Seleção do Portfólio de Projetos**

Dentre as funções de um gestor de portfólio está a seleção, a priorização e o balanceamento do portfólio, visando selecionar os projetos mais vantajosos dentre as várias alternativas, tendo em vista os recursos financeiros disponíveis para esse fim, além de outros fatores internos e externos à organização (KOOROS e MCMANIS, 1998). Esta é uma das decisões mais importantes para a direção de uma organização (KIM, 2006). Porém, a seleção de projetos para compor o portfólio é uma tarefa complexa, que geralmente envolve projetos com diferentes características, riscos e custos (BORGONOVO e PECCATI, 2006). Por isso, os gestores costumam se valer de técnicas que os auxiliam na tomada de decisão (VERBEETEN, 2006), aumentando a probabilidade de cumprirem seus objetivos.

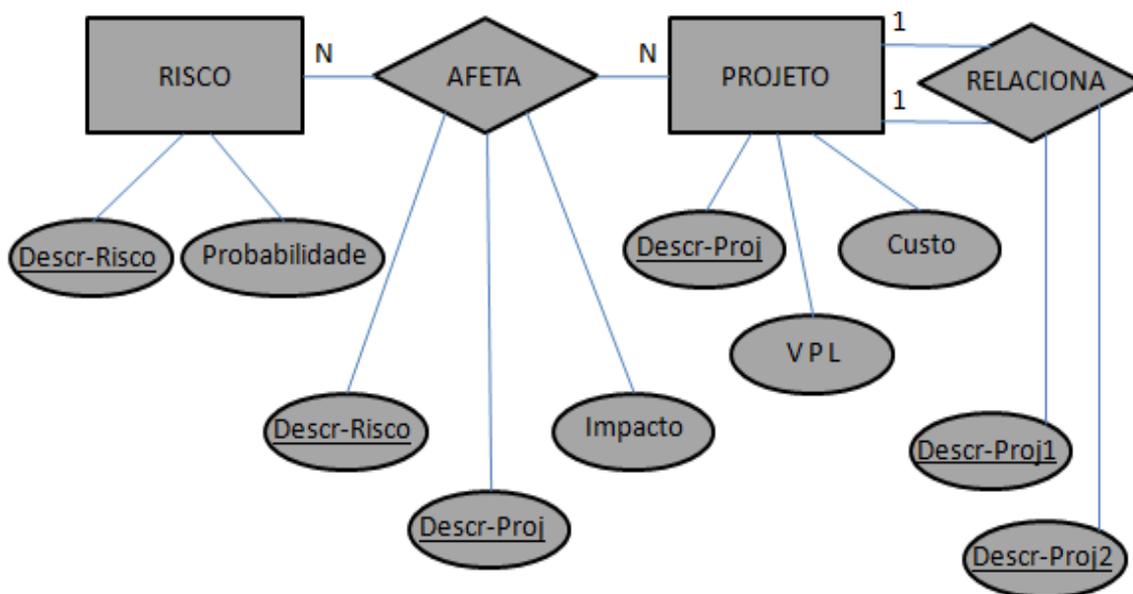
Em ARCHER e GHASEMZADEH (1999), a seleção de projetos para composição de um portfólio é definida como sendo uma atividade periódica que envolve a seleção de

um portfólio a partir das propostas de projetos disponíveis e projetos em andamento e que atinja os objetivos organizacionais estabelecidos de maneira desejável, sem exceder os recursos disponíveis ou violar outras possíveis restrições. Em BELTON e STEWART (2002) é possível ver a definição dessa categoria de problemas como a escolha de um subconjunto de alternativas a partir de um amplo conjunto de possibilidades, levando em consideração não apenas as características das alternativas individualmente, mas também a maneira com que elas interagem, com suas sinergias positivas ou negativas.

Recentemente, foi proposta uma técnica de apoio à seleção de portfólio de projetos de software baseada na Moderna Teoria do Portfólio (COSTA, 2011). A técnica caracteriza os projetos e os riscos envolvidos na composição do portfólio com as seguintes informações: (i) identificação para cada projeto candidato; (ii) custo necessário para sua execução; (iii) retorno esperado para cada projeto, geralmente expresso como valor presente líquido; (iv) nome de cada risco que possa afetar um ou mais projetos; (v) probabilidade de ocorrência de cada risco; e (vi) impacto financeiro que cada risco causará nos projetos afetados por ele. A técnica define também relações entre os projetos, onde os pares de projetos podem ser considerados como: (a) dependentes, se os dois projetos precisam, obrigatoriamente, fazer parte do mesmo portfólio; (b) mutuamente exclusivos, se os dois projetos não podem estar presentes no mesmo portfólio; ou (c) independentes, se não existir relação entre os projetos. As relações entre os projetos são tratadas como restrições, impedindo que certas combinações de projetos sejam consideradas na seleção do portfólio.

A figura 3.1 apresenta os elementos envolvidos no problema e os relacionamentos entre eles. A entidade RISCO representa as incertezas que afetam os projetos, podendo ser considerada uma ameaça ou uma oportunidade, e possui como atributos a descrição da incerteza e a probabilidade que essa incerteza possui de ocorrer. A entidade PROJETO representa o conjunto de todos os projetos candidatos a compor o portfólio e contém os seguintes atributos: (i) nome do projeto; (ii) valor presente líquido do projeto, computado ao final do período observado pelo gestor; e (iii) custo do projeto. Estas entidades se relacionam pelo atributo impacto, que indica o impacto financeiro que será causado no projeto caso o risco em questão ocorra. O impacto será positivo se o risco for uma oportunidade, mas se o risco for uma ameaça, o impacto será negativo. Cada risco poderá interferir de maneira diferente em cada projeto e poderá haver mais de um

risco afetando um mesmo projeto, assim como vários projetos podem ser afetados pelo mesmo risco, muitas vezes com impactos distintos.



**Figura 3.1 – Uma representação dos elementos envolvidos no problema**

Para fins de simplificação da análise, assumimos que os benefícios de todos os projetos candidatos podem ser traduzidos em resultados financeiros e que estes projetos foram submetidos a um filtro inicial quanto ao seu alinhamento com os objetivos estratégicos da organização, suas áreas ou departamentos.

A partir das informações sobre os projetos e riscos, a técnica proposta por (COSTA, 2011) identifica todos os portfólios que podem ser formados por combinações válidas de projetos candidatos, ou seja, combinações que atendam às restrições impostas pelas relações entre projetos e pelo orçamento disponível. Em seguida, a técnica identifica todos os possíveis cenários de riscos, com base na combinação dos riscos que foram identificados, calculando a probabilidade de cada cenário de acordo com a probabilidade de ocorrência dos riscos que o compõem. Para cada cenário de risco, a técnica calcula o retorno esperado e o risco de cada portfólio. O retorno esperado do portfólio é calculado pela soma dos retornos esperados dos seus projetos, enquanto o risco do portfólio é calculado pela composição dos riscos de seus projetos, considerando as correlações entre eles. Na proposta de COSTA (2011), as fórmulas de cálculo de risco e retorno não consideram o peso do projeto no portfólio, como na fórmula original de Markowitz, visto que este peso assume valor zero se o projeto estiver fora do portfólio e um, se participar da composição do portfólio. As correlações são calculadas

de acordo com o impacto dos cenários de risco em cada projeto. Maiores detalhes sobre estes cálculos podem ser encontrados em (COSTA 2011).

Finalmente, a técnica constrói uma Fronteira Eficiente, gerando um gráfico de dispersão bidimensional com o risco agregado e o retorno agregado dos portfólios nos eixos. O risco agregado é calculado pela média dos riscos nos diferentes cenários, ponderada pela probabilidade de ocorrência de cada cenário. Um procedimento similar é utilizado para o retorno. A fronteira eficiente é um gráfico côncavo e os portfólios apresentados no limite superior da fronteira são os mais eficientes, ou seja, aqueles que oferecem o maior retorno por unidade de risco.

Uma limitação da técnica proposta é que o seu tempo de execução cresce exponencialmente com o número de projetos candidatos e de riscos identificados, posto que ela tenta analisar todas as possíveis combinações de portfólios em todas as possíveis combinações de cenários de risco. Desta forma, o espaço de busca pode ser aproximado por  $(2^N * 2^R)$  alternativas, onde  $N$  representa o número de projetos candidatos e  $R$  representa o número de riscos identificados. Portanto, dependendo do número de projetos candidatos, a técnica não conseguirá percorrer todas as possíveis alternativas de composição de portfólios em tempo aceitável. Assim, sugere-se a aplicação de uma busca heurística que consiga retornar uma boa solução para o problema da seleção do portfólio de projetos a partir de uma análise parcial do espaço de busca.

Para viabilizar a aplicação da busca heurística, a Seção 3.4 formaliza um modelo para o problema de seleção do portfólio de projetos. Este modelo será utilizado para descrever os indivíduos componentes da população manipulada pelo algoritmo genético e a função objetivo que será perseguida pelo processo de busca.

### **3.4. Formalização do Problema de Seleção do Portfólio de Projetos**

A proposta deste trabalho é tornar a técnica apresentada na Seção 3.3 aplicável em escala quanto ao número de projetos candidatos ao portfólio, através do uso de uma busca heurística baseada em algoritmos genéticos. Neste sentido, é necessária a modelagem do problema de seleção do portfólio de projetos de software como um problema de otimização, para o qual as técnicas de busca possam encontrar soluções eficientes.

O desenvolvimento desta parte do trabalho deu-se para que fosse criado um modelo matemático que pudesse representar o funcionamento da técnica estudada (COSTA, 2011). A formalização desse modelo visa permitir a implementação de uma ferramenta computacional que auxilie o gestor do portfólio nesta decisão e também: (i) demonstrar como um conjunto de restrições, que de forma geral são comuns no processo de seleção e priorização de projetos, podem ser quantificadas e transformadas em restrições matemáticas; (ii) representar a estratégia da técnica descrita em COSTA (2011), que trabalha com os principais fatores envolvidos na composição de um portfólio de projetos – custo, retorno e risco; e (iii) servir de auxílio para a criação de uma heurística que tornará a implementação desta técnica aplicável em larga escala, ou seja, em cenários onde o espaço de busca não permita a análise de todas as soluções possíveis.

A seguir, passamos a uma descrição formal das entidades e relacionamentos envolvidos na aplicação da técnica apresentada na Seção 3.3.

- Seja  $PC$  o conjunto de todos os projetos candidatos ao portfólio da organização. Cada projeto  $p_i \in PC$  representa um projeto candidato e é descrito pelo nome do projeto, o custo necessário para sua execução e o retorno esperado para o projeto.

$PC = \{ p_i \}, i = 1 .. N$ , onde  $N$  é o número total de projetos candidatos.

$p_i = (\text{nome}, \text{custo}, \text{retorno})$

- Seja  $RI$  o conjunto de relacionamentos de inclusão para cada par de projetos. Cada par de projetos  $RI_h \in RI$  representa a restrição de que o segundo projeto deve fazer parte do portfólio quando o primeiro estiver presente.

$RI = \{ RI_h \}$

$RI_h = (p_1, p_2) / p_1 \in PC \wedge p_2 \in PC$

- Seja  $RE$  o conjunto de relacionamentos de exclusão para cada par de projetos. Cada par de projetos  $RE_h \in RE$  representa a restrição de que o segundo projeto não pode fazer parte do portfólio quando o primeiro estiver presente.

$RE = \{ RE_h \}$

$RE_h = (p_1, p_2) / p_1 \in PC \wedge p_2 \in PC$

- Seja  $RS$  o conjunto dos riscos identificados. Cada risco  $r_j \in RS$  representa uma ameaça ou oportunidade que pode afetar um ou mais projetos candidatos. Cada risco é descrito pelo nome da incerteza a que se refere, a probabilidade dessa incerteza se concretizar e o impacto financeiro que causará em cada projeto afetado por ele.

$RS = \{ r_j \}, j = 1 .. R$ , onde  $R$  é o número de riscos identificados

$r_j = (\text{nome}, \text{probabilidade}, \{ \text{impacto}_i \})$

$0 \leq r_j, \text{probabilidade} \leq 100\%$

$r_j, \text{impacto}_i$  é o impacto que o risco  $r_j$  causará no projeto  $p_i \in PC$ , caso ocorra (zero, se  $p_i$  não é afetado por  $r_j$ ).

- Seja  $PT$  o conjunto de todos os portfólios que podem ser formados pelos projetos candidatos.  $PT$  é um conjunto potência de  $PC$ , ou seja, o conjunto de todos os subconjuntos de  $PC$ . Para cada portfólio, podemos definir seu custo como o somatório dos custos de seus projetos.

$PT = \{ P_z \}$

$P_z \subseteq PC : P_z = (w_1, \dots, w_N) / w_i = 0 \text{ se } p_i \notin P_z; w_i = 1 \text{ se } p_i \in P_z; \forall p_i \in PC.$

Sujeito a  $\forall RI_h \in RI \mid RI_h.p_1 \in P_z \Rightarrow RI_h.p_2 \in P_z$

$\forall RE_h \in RE \mid RE_h.p_1 \in P_z \Rightarrow RE_h.p_2 \notin P_z$

$\text{custo}(P_z) = \sum_{i=1}^N P_z.w_i * p_i \cdot \text{custo}$

- Seja  $PCOR$  o conjunto de todos os possíveis cenários de ocorrência dos riscos.  $PCOR$  é o conjunto potência de  $RS$ , ou seja, o conjunto de todos os subconjuntos de  $RS$ . Cada cenário representa a ocorrência de um conjunto de riscos, sendo possível calcular a probabilidade de que o cenário aconteça (a partir das probabilidades dos riscos nele contemplados) e o valor esperado de cada portfólio no cenário (com base no impacto provocado pelos riscos do cenário nos projetos do portfólio).

$PCOR = \{ PCOR_w \}$

$PCOR_w \subseteq RS : PCOR_w = (w_1, \dots, w_R)$

$/ w_j = 0 \text{ se } r_j \notin PCOR_w; w_j = 1 \text{ se } r_j \in PCOR_w; \forall r_j \in RS.$

$$probCenario(PCOR_z) = \prod_{j=1}^R \begin{cases} 1 - r_j \cdot probabilidade, se PCOR_z \cdot w_j = 0 \\ r_j \cdot probabilidade, se PCOR_z \cdot w_j = 1 \end{cases}$$

$$retorno(P_z, PCOR_z) = \sum_{i=1}^N P_z \cdot w_i * (p_i \cdot retorno + \sum_{j=1}^R PCOR_z \cdot w_j * r_j \cdot impacto_i)$$

Com base na probabilidade de ocorrência de cada cenário e no retorno esperado do portfólio nestes cenários, podemos calcular o retorno esperado do portfólio (ou seja, o retorno em todos os cenários, ponderado pela probabilidade de cada cenário).

$$retorno(P_z) = \sum_{j=1}^{2^R} probCenario(PCOR_j) * retorno(P_z, PCOR_j)$$

O risco do portfólio,  $risco(P_z)$ , é calculado a partir do desvio padrão de cada projeto e das correlações entre estes projetos. O desvio padrão de um projeto é calculado pela média das diferenças entre o retorno esperado do projeto ( $p_i \cdot retorno$ ) e o seu retorno em cada cenário de risco, sendo estas diferenças ponderadas pela probabilidade de ocorrência do cenário. A correlação entre dois projetos é calculada aplicando-se o coeficiente de Spearman (correlação não-paramétrica) (SIEGEL e CASTELLAN, 2006) sobre o par de séries de retorno dos projetos em questão em cada cenário.

Finalmente, o índice de desempenho do portfólio  $IDP(P_z)$  pode ser calculado pela razão entre o retorno esperado e o risco do portfólio.  $IDP(P_z)$  é um índice que demonstra a quantidade de retorno atribuída a um portfólio por unidade de risco, que é uma medida teórica do desempenho do portfólio. Assim, quanto maior este índice, maior o grau de desempenho do portfólio.

$$IDP(P_z) = \frac{retorno(P_z)}{risco(P_z)}$$

O gestor deseja selecionar o portfólio  $P_z^* \in P$  que maximize  $IDP(P_z^*)$  e cujo custo esteja restrito ao montante financeiro disponível. Formalmente, esta seleção pode ser descrita como: selecionar um portfólio  $P_z^*$  onde  $\nexists P_z' / IDP(P_z') > IDP(P_z^*)$ ;  $custo(P_z') \leq OD$ ;  $custo(P_z^*) \leq OD$ , onde OD representa o orçamento disponível da organização para desenvolver os projetos componentes do portfólio. Temos então um problema de otimização, conforme mostrado abaixo:

$$Maximizar IDP(P_z) / P_z \in P$$

$$Sujeito a custo(P_z) \leq OD \quad \forall P_z \in P$$

### 3.5. Proposta de Solução

Conforme citado na Seção 3.2, o problema da seleção do portfólio de projetos pode ser analisado como um problema complexo de otimização combinatória. Por isso, neste trabalho propomos a adoção de um método heurístico, na forma de um algoritmo genético, de maneira que possamos determinar o tempo que esse algoritmo demanda na busca por soluções viáveis dentro do espaço de soluções alternativas geradas por uma instância do problema, sem a análise de todas as soluções alternativas.

A técnica proposta por COSTA (2011) executa nove etapas ao procurar por uma solução: (a) caracteriza os projetos candidatos; (b) caracteriza os riscos; (c) define as relações entre os projetos; (d) define os portfólios alternativos; (e) define os cenários de risco; (f) calcula informações dos portfólios; (g) cria a fronteira eficiente; (h) analisa a variabilidade dos retornos; e (i) define o portfólio.

A nossa proposta de busca heurística com algoritmo genético atua sobre a 6ª e sobre a 9ª etapa da técnica estudada, trabalhando sobre os cálculos e informações acerca dos portfólios e na definição de um único portfólio dentre todas as outras possibilidades.

O algoritmo genético cria, de forma aleatória, uma população inicial de soluções com uma codificação binária para descrevê-las. Cada solução (cromossomo) representa um possível portfólio de projetos que respeita as restrições impostas ao problema, ou seja, a restrição orçamentária e a definição da relação de dependências entre os projetos. É composta de  $N$  bits, onde  $N$  representa o número total de projetos candidatos e um bit (gene) corresponde a um projeto candidato ao portfólio, sendo representado por zero ou um. Um gene com valor zero significa que o projeto correspondente àquela posição não faz parte do portfólio que a solução está representando. O valor um indica que o projeto nessa posição faz parte do portfólio. Por exemplo: para um cenário com os projetos A, B, C, D e E, um portfólio que contenha apenas o primeiro, o terceiro e o quarto projeto, seria representado na forma  $P_z = [1, 0, 1, 1, 0]$ , exemplificando uma solução da qual somente os projetos A, C e D fazem parte.

Com a população inicial criada, o próximo passo é a avaliação de cada solução dessa população através da função objetivo  $IDP(P_z)$ , definida na Seção 3.4. Em seguida, essas soluções são ordenadas de forma decrescente, tomando como base o valor retornado da função objetivo (índice de desempenho do portfólio), fazendo com que as melhores

soluções fiquem no topo da lista de soluções da população, ou seja, as soluções que possuem os maiores índices na avaliação. Com a população inicial ordenada, o processo de reprodução (geração de novas populações) é iniciado com a transmissão das duas melhores soluções (maiores índices) para a próxima população. O algoritmo genético foi configurado com o tamanho da população igual a  $\frac{N}{2}$ , onde N representa o número de projetos candidatos ao portfólio.

As soluções escolhidas para gerar novas soluções são selecionadas pelo método da roleta. Este método representa o operador genético de seleção e é utilizado para prevenir a perda de boas soluções na geração de novas populações.

Após o procedimento de seleção, a operação de cruzamento é executada para garantir a diversidade da população, atuando na prevenção de que algum ponto do espaço de busca das soluções não seja examinado. Para esta operação, foi utilizado o método *single point crossover* (DEB, 2001), que age nas soluções previamente selecionadas, trocando seus genes (projetos) em um ponto determinado. A operação de cruzamento é efetuada sobre cada solução com uma probabilidade de 80%. Esse valor foi conseguido através de testes efetuados através da execução do algoritmo, utilizando várias instâncias com diversos tamanhos, que poderão ser vistos no Capítulo 4.

Após a etapa do procedimento de cruzamento, utilizamos *single point mutation* (DEB, 2001) como operador de mutação, com probabilidade de 2%, valor também escolhido através de testes efetuados com o algoritmo. O operador de mutação visa prevenir que todas as soluções dessa população caiam em um ponto ótimo local.

O processo acima descrito é executado até que o critério de parada seja alcançado. Para este estudo, o critério de parada adotado foi o número de avaliações a serem executadas, que equivale a  $N^2$ .

### **3.6 Conclusão**

Nas formulações apresentadas neste capítulo foi possível ver o funcionamento da técnica estudada (COSTA, 2011) e a estratégia proposta para torná-la aplicável em escala quanto ao número de projetos candidatos ao portfólio. A formulação da técnica estudada é útil para a construção de aplicações que auxiliarão o gestor na tarefa de decidir quais projetos farão parte do portfólio da organização.

No capítulo a seguir, demonstraremos o planejamento e a execução de um estudo experimental que foi projetado para avaliar a proposta apresentada neste capítulo, onde iremos analisar a eficiência e eficácia dessa proposta, comparando-a com estratégias de busca mais simples.

### AVALIAÇÃO DO MÉTODO PROPOSTO

---

#### 4.1. Considerações Iniciais

Conforme citamos no Capítulo 3, o problema da seleção do portfólio de projetos pode ser analisado como um problema de otimização combinatória. Assim, para tratar a limitação da técnica descrita em COSTA (2011) no que se refere ao número de projetos candidatos a formar o portfólio, propomos a utilização de um algoritmo genético, de forma que possamos analisar seus resultados, a fim de determinar o tempo gasto pelo algoritmo na busca por uma solução, dentro do espaço de soluções de uma instância, que respeite as restrições impostas ao problema.

Para verificar se a técnica proposta, baseada em uma heurística complexa, encontra soluções na medida em que o número de projetos aumenta e se essas soluções são de melhor qualidade que aquelas apresentadas por métodos de busca mais simples, projetamos e executamos um estudo experimental que compara a busca heurística proposta com outras duas técnicas de busca. Assim como COSTA (2011), consideraremos o contexto em que uma organização receba verba periódica para investir na composição de seu portfólio de projetos e que o retorno e o custo de cada projeto sejam mensuráveis. Neste Capítulo serão apresentados o projeto e a análise desse estudo experimental.

O projeto experimental utilizado nesta pesquisa trata de dois grupos de instâncias: um conjunto de instâncias aleatórias e outro de instâncias do mundo real, cada qual com diferentes configurações quanto ao número de projetos candidatos a compor o portfólio e quanto ao número de riscos envolvidos no problema (maiores detalhes na Seção 4.3). A partir dessas instâncias, o modelo proposto é aplicado utilizando busca heurística (*Algoritmos Genéticos*) (HOLLAND, 1962), busca não-sistemática (*Random Search*)

(RASTRIGIN, 1963) e busca local multi-partida (*Hill Climbing Random Restart*) (PRADITWONG et al., 2011) (ver Seção 4.4). Neste contexto, as técnicas serão comparadas sob os aspectos de custo computacional (tempo de processamento na busca da solução) e retorno do portfólio selecionado (índice de desempenho do portfólio).

Este Capítulo está dividido em sete seções. A Seção 4.1 apresenta estas considerações iniciais e discute, em linhas gerais, o desenvolvimento do estudo experimental. A Seção 4.2 discute as questões de pesquisa que nortearam o desenvolvimento do estudo. A Seção 4.3 apresenta as instâncias utilizadas para o estudo experimental. A Seção 4.4 apresenta os algoritmos utilizados no estudo, fazendo uma descrição breve sobre os parâmetros utilizados e a forma como foram desenvolvidos. A Seção 4.5 apresenta e discute os resultados do estudo experimental realizado. A Seção 4.6 discute as ameaças à validade do estudo experimental, enfatizando as ações que foram tomadas para minimizá-las e, finalmente, a Seção 4.7 encerra o Capítulo.

## 4.2. Questões de Pesquisa

Com fins de comparar a técnica proposta, baseada em uma heurística complexa, com técnicas de busca mais simples (busca local e busca aleatória), foram consideradas as seguintes questões de pesquisa para este estudo experimental:

**QP1. Eficiência como critério de avaliação.** Qual o custo computacional necessário para executar cada um dos três algoritmos aplicados com a estratégia descrita na técnica de COSTA (2011) em termos do tempo necessário para encontrar soluções em cenários com diferentes números de projetos candidatos a compor o portfólio?

Esta questão de pesquisa compara o tempo de execução de cada algoritmo de busca utilizado no estudo, de forma que se avalie se soluções podem ser encontradas em situações que representem um aumento progressivo no número de projetos disponíveis para formação do portfólio. Nossa expectativa acerca deste estudo experimental é que o algoritmo genético apresente uma média de tempo de execução maior do que a média do Hill Climbing.

Devido a fatores como gerenciamento de memória e programas residentes, que não permitem a homogeneidade de processamento do longo do tempo nos computadores utilizados para execução de estudos experimentais, muitos

pesquisadores costumam medir o custo computacional de um algoritmo de busca através do número de rodadas, avaliações ou ciclos que o algoritmo executa. Por outro lado, por estarmos interessados no tempo que o gestor ou tomador de decisão poderá dispor na espera por informações que apóiem a decisão de formação do portfólio, optamos por trabalhar com o tempo de processamento gasto pelo algoritmo de busca na execução dos estudos experimentais.

**QP2. Eficácia como critério de avaliação.** Dentre os algoritmos utilizados no estudo, qual maximiza o índice de desempenho do portfólio?

Esta questão compara as soluções apresentadas pelos três algoritmos quanto ao valor do índice de desempenho do portfólio, que é a medida maximizada pelo processo de busca pelas melhores soluções (melhores composições de portfólio). Esperamos que as soluções apresentadas pelo algoritmo genético sejam melhores do que as soluções apresentadas pelos outros dois algoritmos de busca, ou seja, que a média dos índices de desempenho dos portfólios selecionados pelo algoritmo genético seja maior do que as médias dos índices dos portfólios selecionados pelos outros dois algoritmos de busca. Esperamos também que o desvio padrão das soluções apresentadas pelo algoritmo genético seja menor do que o desvio padrão das soluções apresentadas pelo Hill Climbing e pelo Random Search.

#### **4.3. Instâncias Utilizadas no Estudo**

Para responder às questões colocadas na Seção 4.2, este estudo experimental utilizou dois grupos de instâncias: instâncias aleatórias e instâncias do mundo real.

O primeiro grupo foi composto por um conjunto de 16 instâncias geradas com dados aleatórios através de um gerador de instâncias desenvolvido pelo grupo de pesquisa, variando-se o número de projetos candidatos ao portfólio (25, 50, 75 e 100) e o número de riscos envolvidos no problema (3, 5, 7 e 10). Entendemos que o número de projetos representa o tamanho da instância, enquanto o número de riscos representa a sua complexidade. Desta forma, as instâncias aleatórias permitem avaliar o comportamento dos algoritmos de busca para diferentes números de projetos, expostos a diferentes conjuntos de riscos. Os números de projetos e riscos em cada instância foram escolhidos

porque estimativas iniciais mostram que a versão original da técnica estudada (COSTA, 2011) não consegue apresentar resultados em um tempo aceitável para cenários que contenham mais do que 20 projetos.

Assumindo uma moeda hipotética, o valor presente dos projetos foi gerado segundo uma distribuição uniforme com limite inferior de \$1.000 e limite superior de \$1.500. O custo dos projetos também seguiu uma distribuição uniforme com limites entre \$15.000 e \$30.000 e a restrição orçamentária de todas as instâncias aleatórias foi fixada em 60% do valor máximo de custo que um portfólio contendo todos os projetos da instância poderia alcançar, de acordo com o maior valor de custo que poderia ser atribuído a cada projeto. Por exemplo, para instâncias com 25 projetos o custo máximo de todos estes projetos seria \$750.000 ( $\$ 30.000 \times 25$  projetos). Neste caso, a restrição orçamentária para esse grupo de instâncias seria o equivalente a 60% de \$ 750.000, que corresponde a \$450.000. A probabilidade de ocorrência dos riscos seguiu uma distribuição uniforme de 0% a 100% e o impacto que cada risco causará em cada projeto seguiu uma distribuição uniforme entre -\$6.800 e \$5.000, onde valores negativos representam ameaças, enquanto valores positivos representam oportunidades. Todos os valores citados acima foram estabelecidos por conveniência para que houvesse consistência entre risco e retorno nas instâncias geradas.

O segundo grupo de instâncias, que serve como um exemplo de problema do mundo real, consiste em um conjunto de informações concedidas por uma organização do setor elétrico, retratando os projetos candidatos ao portfólio de 2011 e os riscos que podem atingir os negócios da organização. Para compor o estudo experimental foram selecionados 250 projetos candidatos a receber investimentos de várias áreas da organização. Esses projetos foram selecionados pelo fato de já possuírem seus custos estimados pelas áreas responsáveis e foram divididos em quatro categorias: (a) implantações, instalações e obras em geral, contendo 25 projetos dos 66 projetos identificados nesta categoria; (b) manutenções, melhorias e modernizações, com 50 projetos dos 315 identificados para esta categoria; (c) Pesquisa e Desenvolvimento, contendo 75 projetos no total; e (d) novos empreendimentos e diversos, com 100 projetos ao todo. O critério utilizado para a escolha dos projetos dentro de cada categoria foi sua data de inclusão no sistema da organização, ou seja, os projetos mais antigos foram escolhidos até o número desejado para compor cada categoria. Este

critério foi utilizado para que não houvesse influência na escolha dos projetos em relação ao custo ou retorno estimado.

Cada categoria representa um centro de custo na organização e possui um orçamento próprio para implementação de um portfólio no período. O somatório destes valores compõe o orçamento total da organização para investimentos no ano de 2011. Como não estamos trabalhando com todos os projetos identificados pela organização (556 projetos), utilizamos um valor proporcional do orçamento destinado a cada categoria de projetos, como mostrado a seguir:

- (a) para a primeira categoria de projetos, a organização destina R\$ 450 milhões. Como estamos trabalhando com 25 projetos, que equivalem a aproximadamente 38% do total de projetos desta categoria, a restrição orçamentária para este grupo foi calculada em 171 milhões (38% do total do orçamento para esta categoria);
- (b) para a segunda categoria, a organização possui um orçamento de R\$ 300 milhões. Como trabalhamos com apenas 16% do total de projetos desta categoria, consideramos uma restrição orçamentária de 48 milhões;
- (c) para a terceira categoria, a organização possui um orçamento previsto de 50 milhões, que foi utilizado em sua totalidade; e
- (d) para a quarta e última categoria de projetos, a organização possui 1,2 bilhões para investir, também utilizado por completo.

Os custos individuais e os retornos estimados para cada projeto candidato foram orçados por funcionários e consultores da organização, não sendo divulgada, por razões de confidencialidade, a estratégia utilizada para chegar nesses valores. Para projetos com duração superior a um ano, consideramos o custo total do projeto e não apenas o custo em determinado período, ou seja, o custo para cada projeto indica o valor total a ser investido para iniciar e terminar o mesmo. Para os retornos estimados, consideramos o retorno anual dos projetos.

Foram também identificados e catalogados 114 diferentes eventos que podem gerar impacto nos negócios da organização. Desse total, 106 estão diretamente relacionados ao portfólio de projetos, podendo acarretar perdas financeiras para a companhia. O levantamento foi realizado por consultores de uma instituição de ensino superior do Rio de Janeiro através de questionários e entrevistas com funcionários de todas as diretorias da organização. Para cada risco, foi estimada sua probabilidade de ocorrência e o

impacto que causaria no conjunto de projetos afetados por esse risco. Baseado nesta informação, consultores financeiros contratados pela organização chegaram aos impactos individuais de cada risco para cada projeto, não sendo divulgada a estratégia utilizada para chegar nesses valores. Como não é o objetivo desta pesquisa tratar a limitação quanto ao número de riscos envolvidos no problema, selecionamos os 10 riscos de maior exposição<sup>2</sup> em relação aos projetos selecionados para executar o estudo experimental.

#### 4.4. Algoritmos Utilizados no Estudo

Neste estudo experimental, além da técnica de busca heurística proposta, baseada em algoritmos genéticos, utilizou-se também as técnicas de Random Search e Hill Climbing para comparação dos resultados. Os algoritmos de busca foram implementados utilizando o framework jMetal (DURILLO et al., 2010), que é uma solução de código reutilizável, orientada a objeto, escrita em Java e criada para dar suporte ao desenvolvimento, à experimentação e ao estudo de algoritmos heurísticos para resolver problemas de otimização.

Foi utilizada uma codificação binária para descrever as soluções para todos os algoritmos. Cada solução representa um possível portfólio de projetos e é composta de N bits, onde N é o número total de projetos candidatos ao portfólio. Por exemplo, para um cenário com os projetos A, B, C, D e E, um portfólio que contenha apenas o primeiro (A), o terceiro (C) e o quarto projeto (D) é representado por  $P_z = [1, 0, 1, 1, 0]$ .

O algoritmo genético foi configurado com o tamanho da população igual a  $N/2$ , onde N representa o número de projetos candidatos ao portfólio. Elitismo foi utilizado para prevenir a perda de boas soluções na geração de novas populações, preservando os dois melhores indivíduos de uma população para outra descendente. Para a operação de cruzamento foi utilizado *single point crossover*, com probabilidade de 80%. Os indivíduos destinados à geração de novos indivíduos foram selecionados pelo método da roleta. Finalmente, utilizamos *single point mutation* como operador de mutação, com probabilidade de 2%. Estes parâmetros foram escolhidos com base em testes de

---

<sup>2</sup> A exposição de um risco é calculada pelo produto da sua probabilidade de ocorrência por seu impacto.

sensibilidade realizados sobre instâncias com pequeno número de projetos. O algoritmo genético executou um número fixo de avaliações da função objetivo, equivalente a  $N^2$ .

Hill Climbing (PRADITWONG et al., 2011) é um algoritmo iterativo que começa com uma solução aleatória no espaço de busca e prossegue tentando encontrar uma solução melhor através de mudanças incrementais, modificando um único *bit* da solução corrente a cada iteração. O algoritmo foi modificado para ser reiniciado em um ponto aleatório quando encontrar um ótimo local, continuando esse reinício aleatório até que atinja o número pré-definido de avaliações da função objetivo do problema. O mesmo número de avaliações da função objetivo utilizado pelo algoritmo genético foi concedido ao Hill Climbing.

Random Search (RASTRIGIN, 1963) é um algoritmo que procura soluções para um problema explorando o espaço de busca de forma aleatória, armazenando a melhor solução até que um critério de parada seja atendido. Por falta de uma sistemática que sirva como guia para o processo de busca, este algoritmo geralmente não apresenta resultados comparáveis com os algoritmos anteriores, mas foi utilizado para confirmar a necessidade de uma heurística na busca de soluções para este problema. Assim, caso o Random Search consiga encontrar regularmente soluções adequadas para o problema, podemos afirmar que não precisamos de algoritmos mais sofisticados para resolvê-lo. Assim como no Hill Climbing, o mesmo número de avaliações da função objetivo utilizado pelo algoritmo genético foi concedido ao Random Search.

Todos os algoritmos de busca envolvem componentes aleatórios. Sendo assim, a comparação entre eles não pode se basear em apenas uma execução de cada um, pois esta execução pode ser beneficiada por um caminho aleatório de qualidade superior à média ou ser prejudicada por um caminho particularmente ruim. Para evitar este efeito, o estudo experimental foi executado 30 vezes<sup>3</sup> para cada algoritmo e cada instância, gerando uma série de valores para o tempo de processamento gasto na execução dos três algoritmos e o índice de desempenho para cada solução. Em seguida, foram calculados a média aritmética e o desvio padrão dos dados de cada série. Finalmente, a comparação das séries geradas pelos três algoritmos de busca foi realizada utilizando o teste de Wilcoxon-Mann-Whitney (FELTOVICH, 2003), que é um teste não-paramétrico de

---

<sup>3</sup> Em algumas execuções dos algoritmos não foi possível encontrar uma solução que atendesse à restrição orçamentária. Nesses casos o ciclo foi re-executado.

inferência estatística utilizado para determinar, entre duas séries de valores, se os resultados podem ser considerados significativamente distintos entre si com 95% de certeza. O teste não-paramétrico foi utilizado porque não podemos afirmar que as séries geradas pelos algoritmos de busca seguem uma distribuição normal homocedástica, condição usualmente requerida por testes estatísticos paramétricos. O teste mostrará se o valor médio dos resultados obtidos pelo algoritmo que apresentou o melhor resultado é significativamente superior ao valor médio dos resultados apresentados pelas outras técnicas de busca.

O código-fonte desenvolvido como parte deste estudo experimental e as instâncias utilizadas para avaliação das técnicas de busca estão disponíveis para *download* em <http://www.uniriotec.br/~marcio.barros/gaportfolio>.

#### **4.5. Análise dos Resultados**

Esta Seção apresenta os resultados do estudo experimental que comparou o desempenho do algoritmo genético com o desempenho dos outros algoritmos de busca apresentados na Seção 4.4 com relação às questões de pesquisa definidas na Seção 4.2. Todas as três estratégias de busca encontraram soluções para as instâncias aleatórias utilizadas no estudo. No entanto, o algoritmo genético maximizou a função objetivo utilizada neste problema e o Hill Climbing utilizou menos tempo de processamento para apresentar suas soluções.

Para tornar o tempo comparável entre as diversas execuções e instâncias utilizadas no estudo, o mesmo computador foi utilizado ao longo de todas as execuções. O computador utilizado para a execução deste estudo possui um processador AMD Athlon II X2 245 2,91 GHz, com 2GB de memória RAM DDR3 e 160 GB de disco rígido de 7200rpm SATA, executando Windows 7, versão de 32 bits e foi utilizado em regime de exclusividade, ou seja, nenhum outro software que não o jMetal (na IDE Eclipse) estava sendo executado durante o estudo.

Esta Seção está dividida em duas partes: a primeira apresenta os resultados coletados a partir do estudo experimental que utilizou três métodos de busca sobre instâncias geradas aleatoriamente. A segunda parte apresenta os resultados coletados a partir do estudo que utilizou os mesmos três métodos de busca, mas sobre instâncias do mundo real.

#### 4.5.1. Resultados do Estudo Experimental sobre Instâncias Aleatórias

A tabela 4.1 apresenta a média aritmética (MED) e o desvio padrão (DP) do tempo de processamento, medido em segundos, para a execução de cada um dos algoritmos utilizados no estudo para cada instância ao longo das várias execuções. A intenção é comparar os algoritmos, de forma que possamos determinar o tempo que cada um deles demanda na busca por uma solução, o que nos ajudará a responder a primeira questão de pesquisa descrita na Seção 4.2.

O teste de Wilcoxon-Mann-Whitney foi aplicado sobre o tempo de processamento do algoritmo de busca que apresentou o menor custo computacional médio, comparando-o com o tempo de processamento dos outros dois algoritmos de busca para saber se os resultados são considerados significativamente distintos com 95% de certeza ( $\alpha = 0.05$ ). Cada resultado significativamente distinto dos outros dois resultados é marcado com negrito na tabela 4.1. No topo da tabela 4.1 está a identificação de cada um dos três algoritmos utilizados.

**Tabela 4.1 – Custo computacional despendido nas instâncias aleatórias**

<i>INSTÂNCIAS</i> <i>Projetos / Riscos</i>	<i>Algoritmo Genético</i> <i>MED / DP</i>	<i>Hill Climbing</i> <i>MED / DP</i>	<i>Random Search</i> <i>MED / DP</i>
025P / 03R	1,0 ± 0,0	1,0 ± 0,0	1,0 ± 0,2
025P / 05R	1,0 ± 0,0	1,0 ± 0,0	1,0 ± 0,0
025P / 07R	1,0 ± 0,0	1,0 ± 0,0	1,0 ± 0,0
025P / 10R	1,3 ± 0,5	1,1 ± 0,3	2,0 ± 0,0
050P / 03R	32,5 ± 0,9	32,6 ± 0,9	42,9 ± 0,3
050P / 05R	24,1 ± 0,7	24,3 ± 1,4	34,9 ± 0,3
050P / 07R	24,0 ± 1,2	<b>23,2</b> ± 1,7	36,3 ± 0,4
050P / 10R	<b>24,7</b> ± 0,7	26,5 ± 1,4	46,8 ± 0,4
075P / 03R	184,8 ± 3,5	<b>179,8</b> ± 7,2	242,6 ± 1,3
075P / 05R	148,8 ± 4,2	<b>143,1</b> ± 5,7	205,1 ± 0,3
075P / 07R	146,7 ± 3,3	<b>135,2</b> ± 6,0	205,0 ± 0,2
075P / 10R	195,9 ± 4,9	194,6 ± 6,4	263,5 ± 2,0
100P / 03R	932,6 ± 17,2	<b>875,7</b> ± 15,7	1098,1 ± 4,2
100P / 05R	502,6 ± 10,6	<b>479,9</b> ± 14,0	674,2 ± 8,9
100P / 07R	590,5 ± 12,0	<b>558,5</b> ± 19,4	717,4 ± 5,4
100P / 10R	874,8 ± 22,3	<b>837,6</b> ± 22,3	1078,9 ± 2,3

Observando a tabela 4.1, vemos que Hill Climbing foi o algoritmo que exigiu o menor tempo de execução na maioria das instâncias, seguido pelo algoritmo genético, com tempos bem próximos dos tempos apresentados pelo Hill Climbing nas menores instâncias (25 e 50 projetos). À medida que a instância cresce em tamanho e complexidade, fica claro o menor custo computacional exigido pelo Hill Climbing na busca por soluções, com significativa diferença das outras duas técnicas.

A tabela 4.2 apresenta a média aritmética dos índices de desempenho dos portfólios (MED) e o desvio padrão desses índices (DP) para todas as repetições de cada algoritmo de busca utilizado no estudo experimental sobre as instâncias aleatórias. Novamente, o teste estatístico foi utilizado para comparar os resultados coletados a partir da execução do algoritmo de busca que apresentou os maiores índices de desempenho do portfólio, comparando-o com o tempo de processamento dos outros dois algoritmos de busca. Valores em negrito são significativamente distintos dos resultados dos outros algoritmos de busca com 95% de certeza.

**Tabela 4.2 – Índices de desempenho dos portfólios obtidos nas instâncias aleatórias**

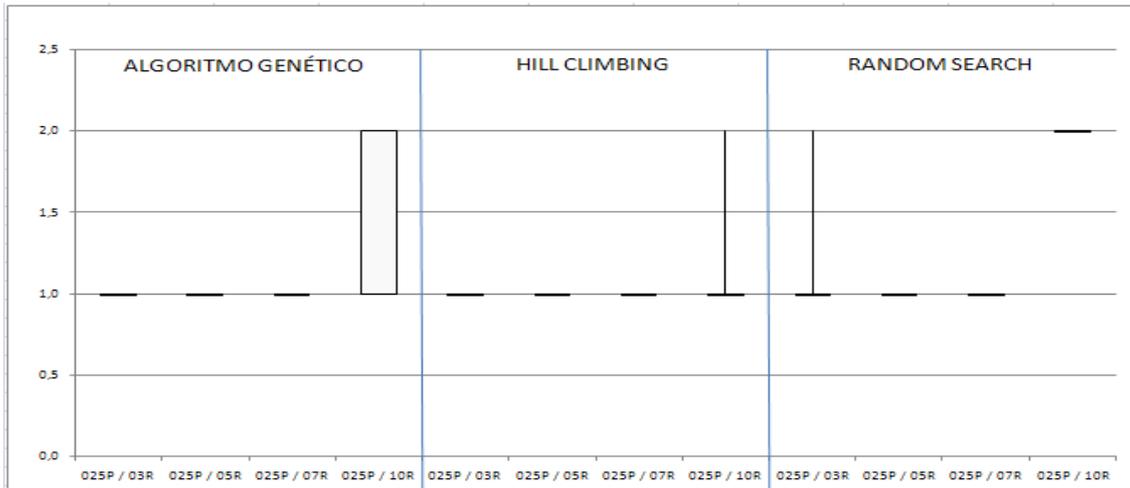
<i>INSTÂNCIAS</i> <i>Projetos / Riscos</i>	<i>Algoritmo Genético</i> <i>MED / DP</i>	<i>Hill Climbing</i> <i>MED / DP</i>	<i>Random Search</i> <i>MED / DP</i>
025P / 03R	<b>4,7</b> ± 1,2	2,9 ± 2,5	0,3 ± 0,4
025P / 05R	6,6 ± 1,6	5,5 ± 2,9	0,2 ± 0,3
025P / 07R	<b>7,7</b> ± 2,6	3,2 ± 5,8	1,2 ± 0,8
025P / 10R	<b>2,5</b> ± 0,1	-0,8 ± 2,3	1,3 ± 0,3
050P / 03R	<b>7,8</b> ± 2,9	2,1 ± 4,8	-0,8 ± 0,1
050P / 05R	<b>8,7</b> ± 1,9	1,5 ± 5,2	-1,7 ± 0,1
050P / 07R	<b>0,6</b> ± 0,2	-2,9 ± 1,9	-2,7 ± 0,1
050P / 10R	<b>0,2</b> ± 0,0	-5,7 ± 4,0	-4,9 ± 0,4
075P / 03R	<b>14,5</b> ± 23,0	4,0 ± 22,5	-2,3 ± 0,1
075P / 05R	5,6 ± 7,8	<b>6,6</b> ± 15,5	-3,3 ± 0,1
075P / 07R	<b>0,4</b> ± 0,3	-1,3 ± 1,5	-1,4 ± 0,1
075P / 10R	5,8 ± 0,1	1,5 ± 4,8	0,8 ± 0,3
100P / 03R	2,8 ± 4,7	7,7 ± 11,0	-0,9 ± 0,1
100P / 05R	4,4 ± 1,6	4,7 ± 6,7	-1,7 ± 0,1
100P / 07R	3,6 ± 1,9	4,1 ± 8,3	-1,9 ± 0,1
100P / 10R	6,4 ± 0,1	1,7 ± 4,6	1,2 ± 0,3

Nos resultados mostrados na tabela 4.2 é possível ver que, em relação à segunda questão de pesquisa (QP2), o algoritmo genético maximiza o índice de desempenho do portfólio, posto que apresenta melhores índices na maioria dos resultados. Podemos notar que o Hill Climbing apresenta uma piora nos resultados quando a complexidade da instância aumenta (maior número de riscos), se distanciando dos resultados do algoritmo genético, como pode ser visto nas instâncias com 7 e com 10 riscos, obtendo alguns resultados piores do que os do Random Search (25P/10R, 50P/07R e 50P/10R). Entretanto, para instâncias maiores (mais projetos) e menos complexas (menos riscos), o Hill Climbing apresenta bons resultados, até superando o resultado do algoritmo genético de forma significativa no caso da instância 75P/5R.

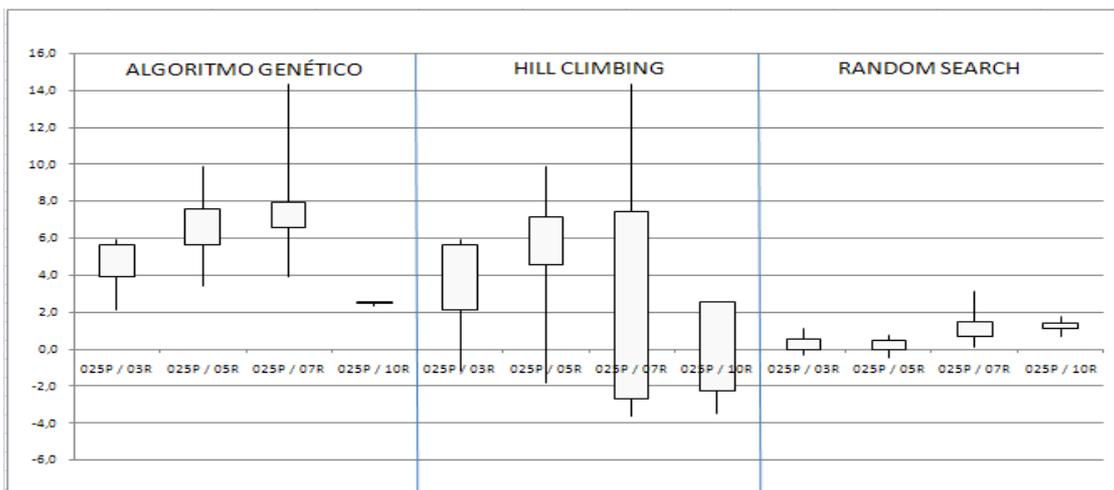
Nas situações em que não há resultados em negrito, não houve significativa diferença apontada pelo teste de Wilcoxon-Mann-Whitney para os valores comparados, como foi o caso das instâncias do grupo de 100 projetos em que, mesmo o Hill Climbing tendo apresentado uma média maior que a média apresentada pelo algoritmo genético, o teste estatístico não mostrou diferença significativa nos resultados.

Algumas vezes o Hill Climbing e o Random Search não conseguiram encontrar soluções adequadas, apresentando soluções com índice de desempenho negativo. Isso indica que haveria prejuízo ao adotar esses portfólios, ou seja, quando o valor total esperado do risco do portfólio excede o retorno esperado para ele.

A seguir, apresentamos os gráficos (*box-plot*) referentes ao custo computacional alcançado e índice de desempenho obtido para cada instância por cada algoritmo de busca. Os gráficos apresentam o primeiro quartil, valor mínimo, valor máximo, mediana e terceiro quartil da série de valores obtidos pelas repetições de cada algoritmo sobre cada instância.



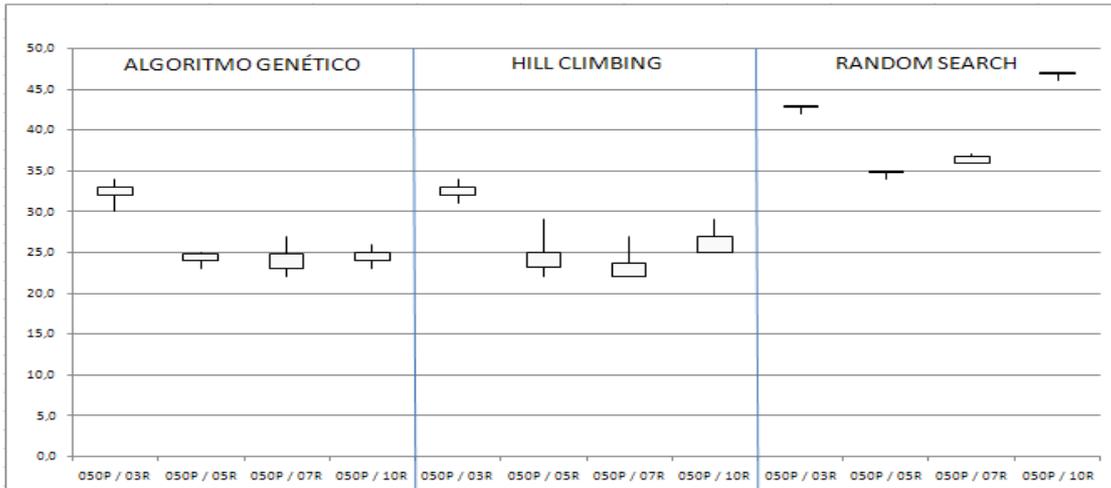
**Figura 4.1 – Custo computacional medido nas instâncias com 25 projetos**



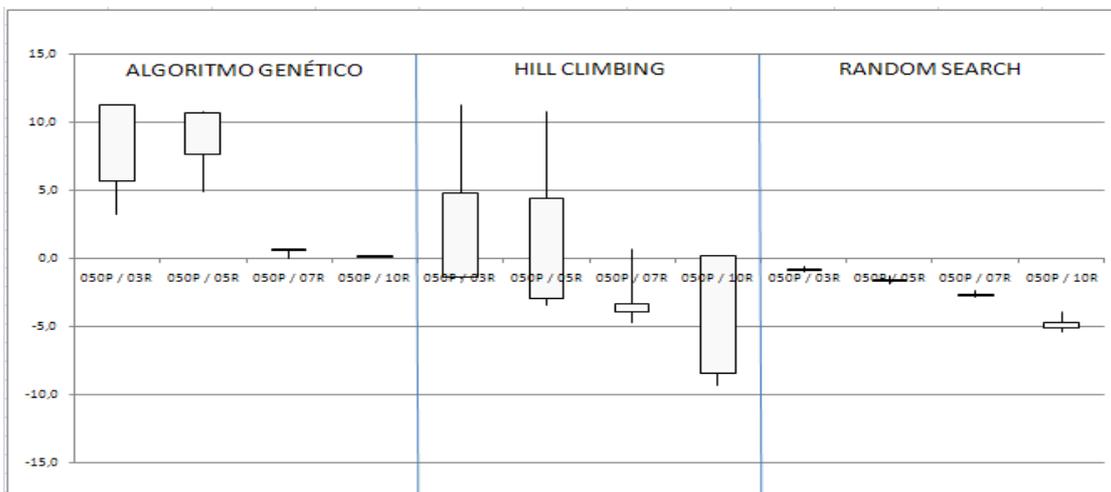
**Figura 4.2 – IDP( $P_z$ ) nas instâncias com 25 projetos**

Os gráficos das figuras 4.1 e 4.2 compreendem todas as instâncias do grupo de 25 projetos e estão divididos em três partes: (a) as primeiras quatro barras são referentes aos resultados apresentados pelo algoritmo genético; (b) as próximas quatro barras são referentes aos resultados apresentados pelo Hill Climbing; e (c) as últimas quatro barras referem-se aos resultados apresentados pelo Random Search.

Os gráficos das figuras 4.3 e 4.4 compreendem todas as instâncias do grupo de 50 projetos e a divisão apresentada segue o modelo dos gráficos das instâncias com 25 projetos.

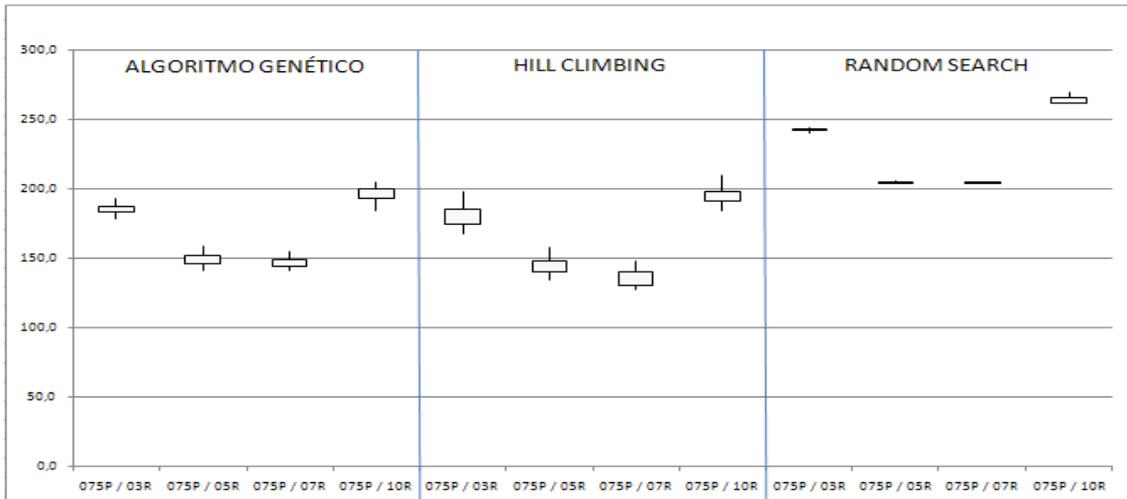


**Figura 4.3 – Custo computacional medido nas instâncias com 50 projetos**

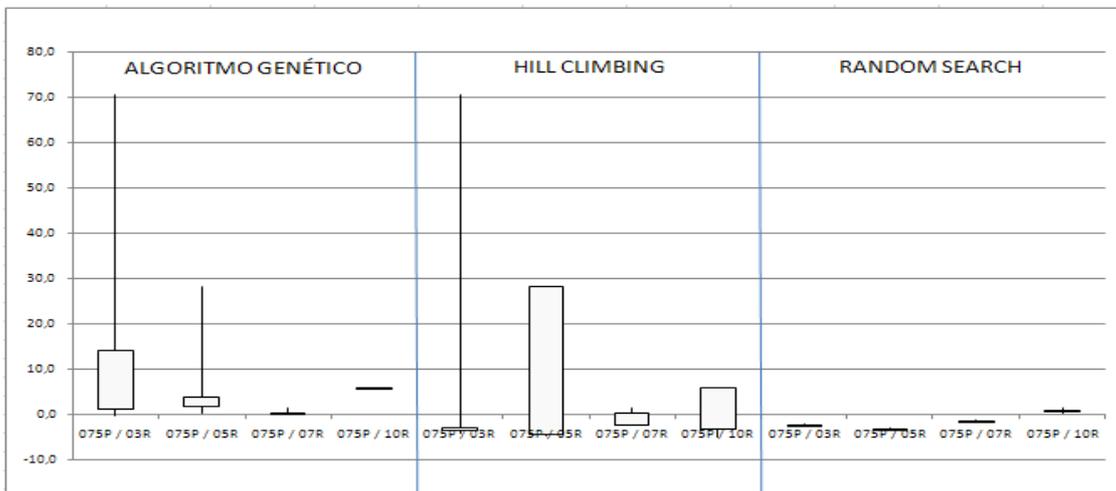


**Figura 4.4 – IDP( $P_z$ ) nas instâncias com 50 projetos**

Os gráficos das figuras 4.5 e 4.6 compreendem todas as instâncias do grupo de 75 projetos e a divisão apresentada nos gráficos segue o modelo dos gráficos das instâncias anteriores.

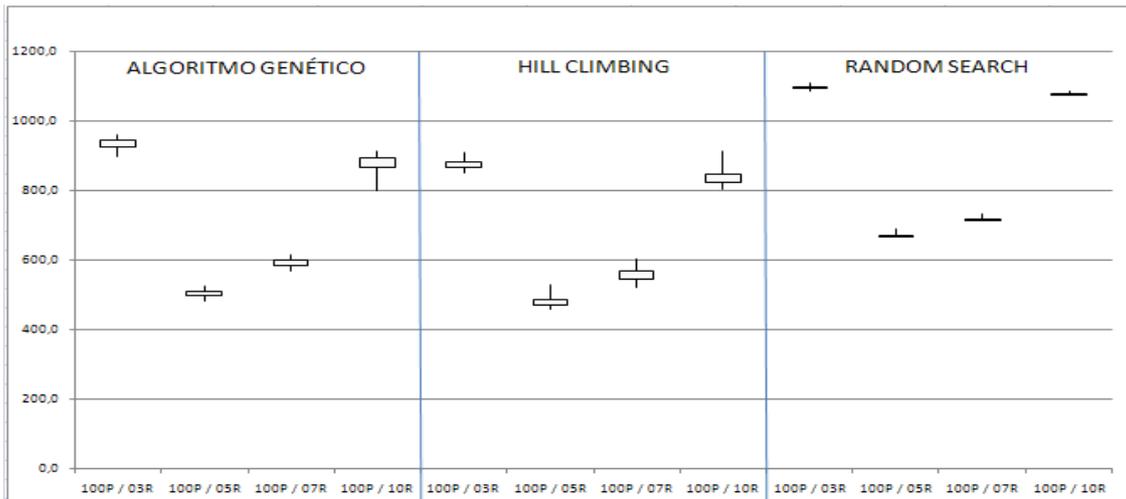


**Figura 4.5 – Custo computacional medido nas instâncias com 75 projetos**

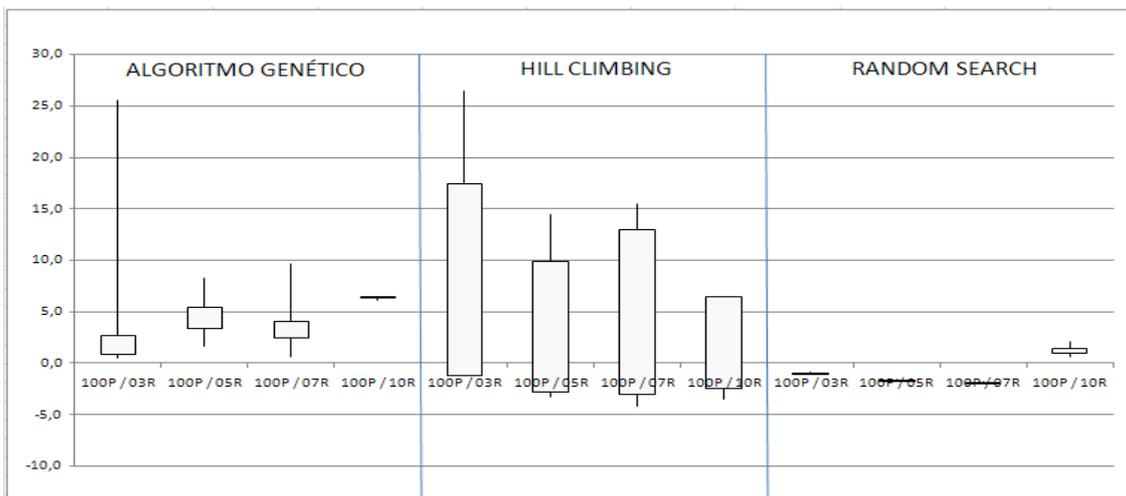


**Figura 4.6 – IDP( $P_z$ ) nas instâncias com 75 projetos**

Os últimos gráficos das instâncias aleatórias (figuras 4.7 e 4.8) compreendem todas as instâncias do grupo de 100 projetos e a divisão apresentada nos gráficos segue o mesmo modelo apresentado nos gráficos anteriores.



**Figura 4.7 – Custo computacional medido nas instâncias com 100 projetos**



**Figura 4.8 – IDP( $P_z$ ) nas instâncias com 100 projetos**

Com os gráficos das figuras 4.1 a 4.8 podemos tirar algumas conclusões. Primeiro, as melhores soluções apresentadas pelo Random Search estão abaixo da média dos índices encontrados pelo algoritmo genético em todas as ocasiões, enfatizando que é necessário o uso de uma busca heurística para resolver o problema de seleção de um portfólio através da técnica descrita em COSTA (2011).

Segundo, embora o custo computacional despendido pelo algoritmo genético seja maior que o do Hill Climbing, a diferença entre esses custos é compensada pela média de melhores soluções quanto ao desempenho do portfólio. Os índices de desempenho dos portfólios selecionados pelo algoritmo genético sugerem que esta heurística é mais

indicada para resolver o problema da seleção de portfólio em cenários com grande número de projetos candidatos. Essa indicação é enfatizada, principalmente, quando se possui um cenário com complexidade crescente, representado pelas configurações com maior número de riscos.

Por fim, podemos notar ainda que o Hill Climbing apresentou alguns valores de pico maiores que os melhores índices encontrados pelo algoritmo genético. No entanto, na média geral o algoritmo genético mostra resultados melhores. Outro ponto notado foi o desvio padrão dos índices de desempenho do portfólio apresentados pelo Hill Climbing, que mostraram maior dispersão da média que o apresentado pela heurística proposta, reforçando ainda mais a qualidade do algoritmo genético quando comparado ao Hill Climbing, pois este último apresenta alguns resultados muito ruins (índices de desempenho do portfólio negativos) em todas as instâncias, enquanto que o algoritmo genético não apresenta soluções com índices negativos. Isso ressalta o fato comentado acima acerca do grau de dispersão dos resultados em relação à média, em que podemos ver resultados com altos índices de desempenho (portfólios considerados bons) e na mesma série de dados outros resultados apontando para índices negativos (portfólios com prejuízos à organização). Isso faz do algoritmo genético uma técnica mais segura para retornar resultados que não ocasionem prejuízos à organização.

#### 4.5.2. Resultados do Estudo Experimental sobre Instâncias do Mundo Real

Da mesma forma da Seção 4.5.2, a tabela 4.3 apresenta a média aritmética (MED) e o desvio padrão (DP) do tempo de processamento, medido em segundos, para a execução de cada um dos algoritmos utilizados no estudo para cada instância do mundo real ao longo das várias execuções. Tal como nos resultados dispostos na tabela 4.1, cada resultado significativamente menor dos outros dois resultados é marcado com negrito.

**Tabela 4.3 – Custo computacional medido nas instâncias do mundo real**

<i>INSTÂNCIAS</i> <i>Projetos / Riscos</i>	<i>Algoritmo Genético</i> <i>MED / DP</i>	<i>Hill Climbing</i> <i>MED / DP</i>	<i>Random Search</i> <i>MED / DP</i>
025P / 10R	1,8 ± 0,4	1,6 ± 0,5	2,0 ± 0,0
050P / 10R	39,0 ± 4,6	38,9 ± 2,8	10,7 ± 0,5
075P / 10R	197,1 ± 8,0	<b>179,6 ± 9,2</b>	280,9 ± 0,4
100P / 10R	954,3 ± 14,5	<b>872,5 ± 17,7</b>	1246,9 ± 11,1

Com relação à questão de pesquisa QP1, os resultados impressos na tabela 4.3 mostram que o algoritmo genético requer mais tempo que o Hill Climbing para encontrar uma solução, mas o Random Search é quem mais demanda tempo de processamento na busca por soluções. Comparando os resultados dos algoritmos sobre as instâncias do mundo real com os resultados sobre as instâncias aleatórias, podemos notar que o Hill Climbing é quem demanda o menor tempo em quase todas as instâncias. No entanto, esse baixo custo computacional vem acompanhado bem de perto pelo algoritmo genético, que ainda apresentou melhores resultados quanto ao índice de desempenho do portfólio em todas as instâncias reais, conforme mostrado na tabela 4.4.

**Tabela 4.4 – Índice de desempenho do portfólio das instâncias do mundo real**

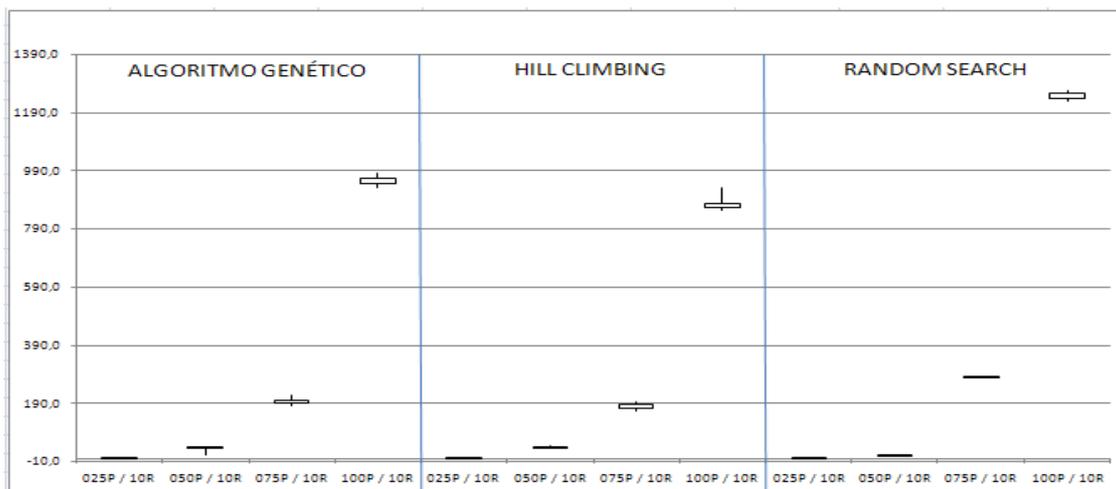
<i>INSTÂNCIAS Projetos / Riscos</i>	<i>Algoritmo Genético MAI / DPI</i>	<i>Hill Climbing MAI / DPI</i>	<i>Random Search MAI / DPI</i>
025P / 10R	4,4 ± 0,0	3,1 ± 2,4	1,58 ± 0,7
050P / 10R	<b>3,7 ± 0,0</b>	2,8 ± 1,8	3,3 ± 0,1
075P / 10R	<b>36,7 ± 0,6</b>	12,1 ± 15,7	11,4 ± 1,6
100P / 10R	10,5 ± 3,0	6,8 ± 8,8	-1,1 ± 0,0

Na tabela 4.4, que apresenta os índices de desempenho dos portfólios das instâncias reais, podemos observar valores mais elevados na terceira e na quarta instância. Isso se deu por causa da natureza dos projetos dessas instâncias, tratando-se de projetos de P&D e projetos de novos empreendimentos, que normalmente envolvem retornos financeiros mais expressivos. Podemos ver também que o algoritmo genético foi capaz de encontrar melhores soluções que os outros dois algoritmos de busca em todas as instâncias. Em duas destas soluções temos 95% de certeza da significância da distinção entre os valores apresentados pelo algoritmo genético e o Hill Climbing, que apresentou melhores soluções que o Random Search em três das quatro instâncias reais.

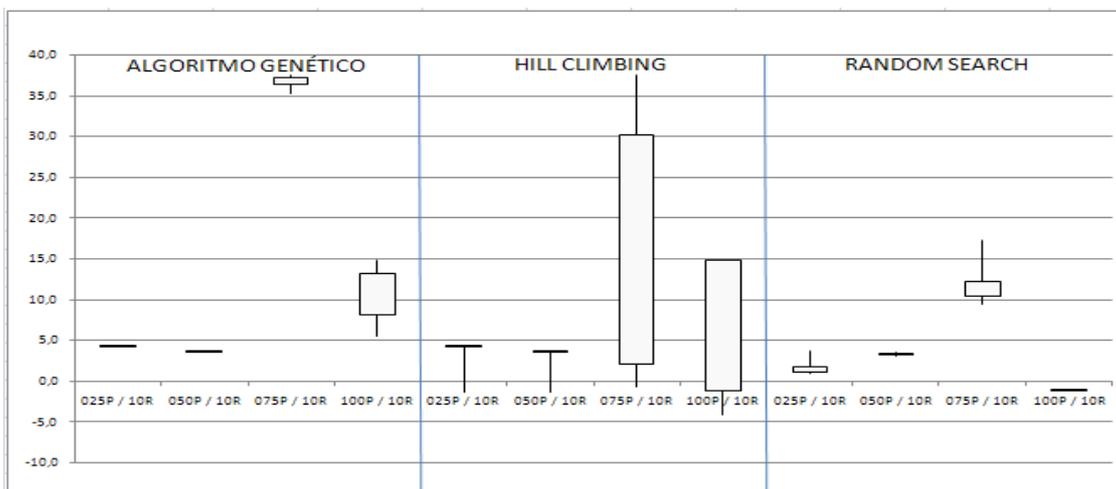
Outro comportamento notado é que o Hill Climbing apresenta resultados ruins nas duas primeiras instâncias, em que o orçamento é bem restrito, assemelhando-se em qualidade ao Random Search. Seriam necessárias mais execuções desses algoritmos com instâncias que variassem o limite orçamentário para podermos afirmar que o Hill Climbing não é recomendado quando o orçamento é reduzido, mas os resultados do estudo apontam nesta direção. Para a questão de pesquisa QP2, podemos sugerir a necessidade de uma busca heurística para encontrar uma solução adequada, posto que o

algoritmo genético encontrou soluções de maior qualidade que as outras técnicas de busca com relação ao valor do índice de desempenho do portfólio.

O gráfico mostrado na figura 4.9 compreende todas as instâncias do mundo real, organizadas da mesma maneira que os gráficos com as instâncias aleatórias. Tal como naqueles gráficos estamos utilizando as informações do primeiro quartil, valor mínimo, valor máximo, mediana e terceiro quartil da série de valores obtidos pelas repetições de cada algoritmo sobre cada instância do mundo real.



**Figura 4.9 – Custo computacional medido nas instâncias do mundo real**



**Figura 4.10 – IDP( $P_z$ ) nas instâncias do mundo real**

Nas figuras 4.9 e 4.10 podemos ver comportamento semelhante aos das instâncias aleatórias, onde o algoritmo genético apresenta um custo computacional bem próximo do custo apresentado pelo Hill Climbing e se repete o fato do Hill Climbing apresentar alguns resultados negativos nas execuções. Tal como evidenciado na tabela 4.4, a média dos índices de desempenho do portfólio apresentados pelo algoritmo genético é superior, com sua pior solução sempre acima do melhor resultado apresentado pelo Random Search. Com isto, fica evidenciado que se faz necessária a utilização de uma heurística para resolver o problema da seleção do portfólio de projetos utilizando a técnica descrita em COSTA (2011).

Notamos que, para as instâncias reais utilizadas neste estudo experimental, o algoritmo genético convergia recorrentemente para a mesma solução. Presumimos que esse resultado foi alcançado devido à forma como as correlações foram geradas para estas instâncias, que acabou resultando em projetos fortemente correlacionados. Por estar fora do contexto desta dissertação, não efetuamos uma investigação mais profunda sobre este aspecto, que pode indicar insights sobre como as empresas realizam suas análises de risco.

#### **4.6. Ameaças à Validade do Estudo**

Segundo WOHLIN et al. (2000), em todo estudo experimental existem ameaças à validade das conclusões que poderiam ser obtidas e, portanto, alguns cuidados precisam ser tomados. Os autores citam quatro categorias de ameaça à validade de um estudo experimental: Interna, Externa, Construção e Conclusão.

Recentemente, BARROS e DIAS-NETO (2011) publicaram um relatório técnico que aborda as ameaças à validade de estudos empíricos em Engenharia de Software baseada em Buscas. Este relatório apresenta diferentes situações específicas de estudos neste campo da Engenharia de Software que podem resultar em vieses ou ameaças às conclusões obtidas de um estudo experimental.

Seguindo essa abordagem, podemos considerar ameaças externas: (i) a falta de uma definição clara das instâncias utilizadas no estudo; (ii) a não apresentação de uma estratégia para seleção das soluções do espaço de busca; e (iii) a falta de instâncias de tamanho e complexidade crescentes para a avaliação da abordagem proposta. Estas

ameaças foram tratadas neste estudo com a descrição, na Seção 4.3, dos dados de geração das instâncias e das estratégias de seleção adotadas na Seção 4.4. Ao utilizarmos instâncias de quatro tamanhos distintos, no que se refere aos projetos candidatos ao portfólio, e com quatro diferentes medidas de complexidade, referindo-se ao número de riscos envolvidos no problema, tratamos o último item das ameaças externas.

As ameaças internas incluem: (i) a não discussão do código fonte utilizado no estudo experimental; (ii) a falta de informação quanto aos parâmetros utilizados na execução das técnicas de busca empregadas; (iii) a falta de uma descrição clara de procedimentos para levantamento de dados; e (iv) a falta de exemplos do mundo real. Para o tratamento destas ameaças, foram disponibilizadas para download no endereço eletrônico descrito na Seção 4.4, as instâncias geradas para avaliação das técnicas propostas e o código-fonte desenvolvido como parte deste estudo experimental. Descrevemos também a parametrização utilizada na execução de todas as três técnicas de busca utilizadas no estudo, onde informamos a forma de codificação para representarmos as soluções, o tamanho da população empregada e o número de avaliações que serviu como critério de parada. Os operadores utilizados e a probabilidade de seu uso também foram descritos no texto, bem como o número de repetições para cada algoritmo, além da especificação do teste de inferência estatística aplicado nas séries de resultados geradas. Descrevemos ainda a maneira com que geramos as instâncias aleatórias utilizadas no estudo e utilizamos exemplos do mundo real, coletados em uma grande empresa do setor elétrico, contendo projetos de diversos segmentos da empresa.

Ameaças à validade da construção de estudos experimentais podem ser: (i) a falta de uma avaliação da métrica de custo, tratada neste artigo como sendo a resposta da primeira questão de pesquisa (QP1), onde foi usada uma métrica aceitável para estimar o custo de cada algoritmo (tempo de execução), devidamente justificada na Seção 4.2; (ii) a falta de uma avaliação da validade das métricas de eficácia, discutido no Capítulo 3 desta dissertação, com a definição do  $IDP(P_z)$ ; e (iii) a falta de uma discussão sobre o modelo que suporta o processo de otimização, discutido também no Capítulo 3, ao modelar o problema da seleção de portfólio de projetos como um problema de otimização.

Por fim, as ameaças às conclusões incluem: (i) a desconsideração da natureza aleatória dos algoritmos de busca heurística; (ii) a falta de uma base de comparação relevante para avaliar um algoritmo de busca heurística; e (iii) a falta de testes de inferência estatística. Estas ameaças foram tratadas neste estudo através da execução de cada algoritmo sobre cada instância por várias vezes (no mínimo 30 execuções), apresentando a média aritmética e o desvio padrão dos  $IDP(P_z)$ . Segundo BRIAND e ARCURI (2011), recomenda-se executar o algoritmo usado no teste por, pelo menos, 1000 vezes sobre cada instância para que se consiga poder estatístico nos resultados. Porém esse alto número de execuções se tornaria inviável devido ao tempo de cada execução dos algoritmos sobre as maiores instâncias, conforme vemos na tabela 4.1.

Quanto ao segundo item das ameaças às conclusões, além da heurística proposta, utilizamos neste trabalho um algoritmo de busca não sistemática, para que pudéssemos provar que este problema não pode ser resolvido adequadamente com a seleção de uma solução aleatória no espaço de busca de soluções. Utilizamos também um algoritmo de busca sistemática mais simples que a nossa proposta, tudo isso com o intuito de provar que uma heurística complexa se faz necessária para se obter soluções viáveis para o problema tratado.

O terceiro item das ameaças às conclusões foi tratado através da comparação dos resultados apresentados pelos algoritmos usando o teste estatístico não-paramétrico de Wilcoxon-Mann-Whitney.

#### **4.7. Conclusão**

Neste Capítulo foi possível ver a apresentação de um estudo experimental para comparação dos resultados apresentados pelas abordagens de busca heurística, de busca local multi-partida e de busca não-sistemática, quanto à eficácia das soluções encontradas para o problema da seleção do portfólio de projetos, bem como o custo computacional despendido por esses algoritmos.

Foi possível confirmar, através desse estudo experimental, que a técnica de apoio à seleção de portfólio de projetos de software baseado na Moderna Teoria do Portfólio (COSTA 2011), aliada a alguma das três estratégias de busca utilizadas nesse estudo, pode ser utilizada em cenários com variado número de projetos candidatos ao portfólio da organização. Foi observado também que a heurística proposta apresenta melhores

resultados que as outras estratégias de busca utilizadas na pesquisa, pois encontrou os maiores índices de desempenho do portfólio.

No próximo capítulo apresentamos as conclusões tiradas das análises dos resultados apresentados no estudo experimental e fazemos as considerações finais sobre a pesquisa aqui desenvolvida, listando as contribuições e propostas para futuras pesquisas.

# CONCLUSÕES

---

### 5.1. Considerações Gerais

Neste trabalho de pesquisa, apresentamos uma modelagem do problema de seleção do portfólio de projetos de software como um problema de otimização, o funcionamento da técnica de apoio à seleção de portfólio de projetos de software baseado na Moderna Teoria do Portfólio (COSTA, 2011) e a estratégia proposta para torná-la aplicável em larga escala quanto ao número de projetos candidatos ao portfólio. Inserido no contexto da engenharia de software baseada em buscas (no inglês, Search Based Software Engineering - SBSE), este trabalho se propôs a validar a proposta da busca por soluções viáveis para o problema tratado, nos permitindo fazer a seguir, algumas considerações acerca dos resultados apresentados no estudo experimental.

O custo computacional despendido pela heurística proposta na busca por soluções viáveis para o problema da seleção do portfólio de projetos de software não foi o menor apresentado, como já era esperado. Por outro lado, a heurística proposta apresentou melhores resultados que as outras estratégias de busca utilizadas na pesquisa, sob o aspecto do maior valor retornado pela relação risco/retorno do portfólio, maximizando a função objetivo que trata do índice de desempenho do portfólio.

Em quase todos os resultados, o desvio padrão do índice de desempenho do portfólio apresentado pela heurística proposta (algoritmo genético) ficou menor que o desvio padrão desse índice apresentado pelos métodos de busca mais simples, demonstrando que houve certo padrão na busca pelas soluções, onde grande parte das soluções

apresentadas para cada instância possuía valores aproximados, conforme pode ser visto nos gráficos apresentados no Capítulo 4.

Assim, com a análise dos resultados do estudo experimental realizado, concluímos que a técnica proposta por COSTA (2011) associada a uma heurística não trivial, pode ser utilizada em cenários com elevado número de projetos candidatos ao portfólio, onde consegue encontrar soluções viáveis sob o aspecto da relação risco/retorno, sem avaliar todas as possibilidades alternativas de combinação entre os projetos.

## **5.2. Contribuições**

Além do conhecimento absorvido durante a pesquisa, alguns dos benefícios obtidos com esta dissertação foram:

- O tratamento da limitação da técnica proposta por COSTA (2011), tornando-a aplicável em larga escala quanto ao número de projetos candidatos a formar o portfólio da organização;
- O planejamento e a execução de um estudo experimental para análise dos resultados apresentados pelas técnicas de busca utilizadas na pesquisa, de forma que se pudesse comparar esses resultados e confirmar a necessidade do uso de uma heurística associada à técnica de COSTA (2011) para encontrar soluções viáveis em cenários com grande número de projetos candidatos ao portfólio;
- O desenvolvimento de uma ferramenta computacional para suportar o estudo experimental, cujos parâmetros, código-fonte e instâncias utilizadas estão disponíveis em um endereço eletrônico na Internet (Seção 4.4 do Capítulo 4), de forma que a repetição do estudo seja facilitada a outros pesquisadores, uma vez que uma única execução de um estudo experimental nunca oferece a resposta final para uma questão, tornando-se importante facilitar sua repetição (WOHLIN et al., 2000);
- A modelagem formal do problema de seleção do portfólio de projetos de software como um problema de otimização oferece a vantagem de podermos aplicar outros algoritmos neste mesmo modelo, variando apenas poucos termos da função objetivo e permitindo a outros pesquisadores a comparação dos resultados aqui apresentados pela heurística proposta com diversas outras técnicas existentes na literatura;

- A estratégia descrita na técnica de COSTA (2011) foi aplicada em instâncias reais de uma grande organização do setor elétrico, contendo projetos de diversos segmentos da empresa, denotando contextos diferentes dos contextos de projetos de software para o qual a técnica foi originalmente desenvolvida, demonstrando à comunidade científica que não notamos diferença na aplicação dessa técnica em diferentes domínios de conhecimento.

### **5.3. Limitações**

A abordagem apresentada nesta pesquisa utiliza a estratégia proposta por COSTA (2011) para buscar soluções viáveis dentro do espaço de soluções de uma instância, fazendo com que todas as limitações existentes na técnica estudada, sejam herdadas por ela. Assim, iremos nos focar apenas nas limitações identificadas no âmbito de nosso trabalho de pesquisa e discuti-las a seguir.

- Consideramos que a formulação de modelos matemáticos procura sempre refletir e representar o mundo real. No entanto, fazer com que uma formulação reflita a realidade do problema da seleção do portfólio de projetos de software não é tarefa trivial, pois envolve várias análises, tais como: necessidades do negócio, problemas corporativos, demandas dos clientes, objetivos estratégicos, incertezas, estimativas de retorno, dentre outros, acumulando-se em muitas variáveis que acabam por deixar o modelo formulado simplificado demais pelo pesquisador. Em nossa pesquisa, tratamos apenas o aspecto referente às incertezas identificadas no cenário, as estimativas de retorno dos projetos e os custos de implementação dos portfólios;
- Outra limitação identificada é acerca das boas práticas recomendadas em estudos experimentais para a Engenharia de Software baseada em Buscas, onde cada algoritmo deve ser executado por, pelo menos, 1000 vezes para que se obtenha poder estatístico nos resultados de forma a apoiar a conclusão do pesquisador (BRIAND e ARCURI, 2011). Na presente pesquisa, em virtude do tempo médio de execução de cada ciclo, optamos por executar apenas 30 ciclos para cada instância;
- O estudo experimental concentrou-se apenas na viabilidade de aplicação da proposta quanto ao índice de desempenho do portfólio selecionado e o custo computacional exigido pelo algoritmo, enquanto que outras informações poderiam ser úteis ao

gestor, como, por exemplo, os resultados dos algoritmos sobre instâncias submetidas a uma restrição orçamentária crescente;

- Finalmente, o estudo experimental executou os algoritmos com a população inicial escolhida de forma aleatória, sem que houvesse a participação de um gestor para informar suas preferências na forma de uma solução inicial para os algoritmos. Com isso, não foi possível incorporar a experiência e conhecimento do gestor no modelo.

#### **5.4. Sugestões para Futuras Pesquisas**

A pesquisa contida nesta dissertação não esgotou suas possibilidades e diversas frentes de pesquisa continuam abertas. Muitas limitações ainda subsistem e esperamos que o uso continuado da proposta apresentada faça surgir outras possibilidades de pesquisa. Dentre algumas possibilidades para pesquisa, podemos sugerir:

- Melhorar o modelo proposto, no sentido de aproximar sua formulação matemática à realidade com o desenvolvimento de outros modelos para o problema da seleção do portfólio de projetos de software ou com a incorporação de outras restrições;
- Repetir o estudo experimental com um número maior de repetições e de forma mais abrangente, para permitir uma avaliação da proposta com outras heurísticas, como por exemplo, o GRASP (RESENDE e RIBEIRO, 2003);
- Executar outro estudo experimental que aborde os resultados dos algoritmos de busca utilizados quanto ao grau de restrição orçamentária incidente em cada instância, a fim de comparar qual técnica de busca é a mais indicada quando o orçamento disponível diminui ou aumenta significativamente;
- Realizar estudos experimentais para analisar outras abordagens do problema, como a indicação pelo gestor de uma solução conhecidamente satisfatória para o algoritmo de busca, tornando possível incorporar sua experiência e conhecimento ao modelo;

Por fim, gostaríamos de ressaltar a nossa satisfação devido à grande oportunidade de produzir esse estudo científico na busca por estratégias mais efetivas e tecnologicamente interessantes de otimização, contribuindo com uma melhora incremental nessa área de conhecimento que afeta de forma tão significativa o orçamento de uma organização.

## REFERÊNCIAS BIBLIOGRÁFICAS

---

- ARCHER, N. P., GHASEMZADEH, F., 1999, “An Integrated Framework for Project Portfolio Selection”, *International Journal of Project Management*, v. 17, n. 4 (Aug), pp. 207-216.
- BAKAR, N. A., ADAM, N. L., HASSAN S., 2009, “GA Based Portfolio Optimization Technique”. *Conference on Scientific & Social Research (CSSR 08'09)*, Melaka, Malaysia, 14-15 March.
- BARROS, M. O., DIAS-NETO, A. C., 2011, *Threats to Validity in Search-based Software Engineering Empirical Studies*. In: Relatórios Técnicos do DIA (RelaTe-DIA), Vol. 5, Nº 1, Universidade Federal do Estado do Rio de Janeiro (UNIRIO), Rio de Janeiro, RJ.
- BELTON, V.; STEWART, T. J., 2002, *Multiple Criteria Decision Analysis*, Massachusetts, Kluwer Academic Publishers.
- BORGONOVO, E.; PECCATI, L., 2006, “The Importance of Assumptions in Investment Evaluation”, *International Journal of Production Economics*, v. 101, Issue 2 (Jun), pp.298-311.
- BRIAND, L.; ARCURI, A., 2011, “A Practical Guide for Using Statistical Tests to Assess Randomized Algorithms in Software Engineering”, 33<sup>rd</sup> *International Conference on Software Engineering*, Honolulu , HI, USA, 21-28 May.
- CHANG, T.; YANG, S.; CHANG, K., 2009, “Portfolio optimization problems in different risk measures using genetic algorithm”, *Expert Systems with Applications: An International Journal*, v. 36, Issue 7 (Sep), pp. 10529-10537.
- CHANG, T.; MEADE, N.; BEASLEY, J. E.; SHARAIHA, Y. M., 2000, “Heuristics for Cardinality Constrained Portfolio Optimization”, *Computers and Operations Research*, v. 27, Issue 13 (Nov), pp. 1271–1302.
- COOPER, R. G.; EDGETT, S. J.; KLEINSCHMIDT, E. J., 2001, *Portfolio Management for New Products*. 2 ed. New York, Perseus Publishing.

- COSTA, H. R., 2011, *Apoio à Seleção de Portfólio de Projetos de Software baseado na Moderna Teoria do Portfólio*. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- DEB, K., 2001, *Multi-Objective Optimization using Evolutionary Algorithms*. New York, John Wiley & Sons, Inc.
- DURILLO, J. J.; NEBRO, A.J.; ALBA, E., 2010, “The jMetal Framework for Multi-Objective Optimization: Design and Architecture”. *Conference on Evolutionary Computation (CEC 2010)*, C-7272, Barcelona, Spain, 18-23 July.
- ENGELS, M., 2004, *Portfolio Optimization: Beyond Markowitz*. M.Sc. Thesis, Mathematisch Instituut, Universiteit Leiden, Leiden, The Netherlands.
- EPM CONNECT, 2007, *The Business Exchange for EPM Partner Solution and Services*. In: Report UMT Consulting Group.
- FELTOVICH, N., 2003, “Nonparametric Tests of Differences in Medians: Comparison of the Wilcoxon–Mann–Whitney and Robust Rank-Order Tests”. *Experimental Economics Journal*, Houston, Springer Netherlands, pp. 6-273.
- GEN, M.; CHENG, R., 1997, *Genetic Algorithms & Engineering Design*. New York, Wiley Interscience Publication.
- GOLDBERG, D. E., 2009, *Genetic Algorithms: in Search, Optimization & Machine Learning*. Indiana, RR Donnelley Crawfordsville.
- GOLDBERG, M. C., 2000, *Otimização Combinatória e Programação Linear: Modelos e Algoritmos*. Rio de Janeiro, Editora Campus.
- HOLLAND, J. H., 1962, "Outline for a Logical Theory of Adaptive Systems". *Journal of the Association Computing Machinery (JACM)*, v. 9, n. 3 (Jul), pp. 279–314.
- JOBSON, J. D.; KORKIE, B., 1980, “Estimation for Markowitz Efficient Portfolios”. *Journal of the American Statistical Association*, v. 75, n. 371 (Sep), pp. 544–554.
- KIM, D., 2006, “Capital Budgeting for new projects: on the role of auditing in information acquisition”. *Journal of Accounting & Economics*, v. 41 (Jun), pp. 257-270.

- KONNO, H; WIJAYANAYAKE, A., 2001, “Portfolio optimization problem under concave transaction costs and minimal transaction unit constraints”. *Mathematical Programming*, v. 89, série B (Jul), pp. 233–250.
- KOOROS, S. K.; MCMANIS, B. L., 1998, “A multi-attribute optimization model for strategic investment decisions”. *Canadian Journal of Administrative Sciences*, v. 15, n. 2 (Jun), pp.152-164.
- LAI, K. K.; YU, L.; WANG, S.; ZHOU, C., 2006, “A Double-Stage Genetic Optimization Algorithm for Portfolio Selection”. *13<sup>th</sup> International Conference Neural Information Processing (ICONIP 2006)*, Part III, v. 4234 (Oct), pp. 928 – 937.
- LIN, C.; LIU, Y., 2008, “Genetic Algorithms for Portfolio Selection Problems with Minimum Transaction Lots”. *European Journal of Operational Research*, v. 185, Issue 1 (Feb), pp. 393-404.
- LIN, C.C., 2004, “A Weighted Max–Min Model for Fuzzy Goal Programming”. *Fuzzy Sets and Systems*, v. 142, Issue 3 (Mar), pp. 407–420.
- LACHTERMACHER, G., 2004, *Pesquisa Operacional na Tomada de Decisões*. 2 ed. Rio de Janeiro, Editora Campus.
- MA, J.; FAN, Z.; JIANG, Y.; MAO, J.; MA, L., 2006, “A Method for Repairing the Inconsistency of Fuzzy Preference Relations”. *Fuzzy Sets and Systems*, v. 157, Issue 1 (Jan), pp. 20-33.
- MANSINI, R.; SPERANZA, M. G., 1999, “Heuristic Algorithms for the Portfolio Selection Problem with Minimum Transaction Lots”. *European Journal of Operational Research*, v. 114, Issue 2 (Apr), pp. 219–233.
- MANSINI, R.; SPERANZA, M. G., 2005, “An exact approach for portfolio selection with transaction costs and rounds”. *Journal of the Institute of Industrial Engineers (IIE Transactions)*, v.37, Issue 10 (Oct), pp. 919–929.
- MANSINI, R.; SPERANZA, M. G., 1997, *On Selection a Portfolio with Fixed Costs and Minimum Transaction Lots*. Report n. 134, Dip. Metodi Quantitativi, University of Brescia, Italy.

- MARKOWITZ, H. M., 1952, "Portfolio Selection". *The Journal of Finance*, v. 7 (Mar), pp. 77-91.
- MICHALEWICZ, Z., 1996, *Genetic algorithm + data structures = evolution programs*, 3º ed. New York, Springer-Verlag.
- MOTA, W. L., 1996, *Análise Comparativa de Algoritmos Genéticos para o Problema de Programação de Operações Flow Shop Permutacional*. Dissertação de M.Sc., EESC/USP, São Carlos, SP, Brasil.
- MR MPS.BR., 2011, *Modelo de Referência para Melhoria de Processo do Software Brasileiro. Guia de Implementação - Parte 5: Fundamentação para a Implementação do Nível C do MR MPS-BR*, São Paulo, Associação para Promoção da Excelência do Software Brasileiro – SOFTEX.
- NETTO, F. C., 2010, *Realocação de Tarefas para Resolução de Defeitos em Sistemas de Larga Escala*. Dissertação de M.Sc., Departamento de Informática Aplicada, Universidade Federal do Estado do Rio de Janeiro (UNIRIO), Rio de Janeiro, RJ, Brasil.
- PADOVANI, M., 2007, *Apoio à Decisão na Seleção do Portfólio de Projetos / Uma Abordagem Híbrida Usando os Métodos AHP e Programação Inteira*. Dissertação de M.Sc., Departamento de Engenharia de Produção, Escola Politécnica da Universidade de São Paulo, São Paulo, SP, Brasil.
- PAPADIMITRIOU, C. H.; STEIGLITZ, K., 1998, *Combinatorial Optimization: Algorithms and Complexity*. New York, Dover Publications.
- PMI, 2008, *A guide to the project management body of knowledge (PMBOK Guide)*. 4 ed. Pennsylvania, Project Management Institute, Inc.
- PRADITWONG, K.; HARMAN, M.; YAO, X., 2011, "Software Module Clustering as a Multi-Objective Search Problem". *IEEE Transactions on Software Engineering*, v. 37, n. 2 (Mar), pp. 264-282.

RABECHINI JR., R.; MAXIMIANO, A. C. A.; MARTINS, V. A., 2005, “A Adoção de Gerenciamento de Portfólio como uma Alternativa Gerencial: O Caso de uma Empresa Prestadora de Serviço de Interconexão Eletrônica”. *Associação Brasileira de Engenharia de Produção – (Produção [online])*, v. 15, n. 3 (Dez). pp. 416-433. Disponível em:

<[http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S0103-65132005000300011&lng=pt&nrm=iso](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-65132005000300011&lng=pt&nrm=iso)>

RASTRIGIN, L. A., 1963, "The Convergence of the Random Search Method in the Extremal Control of a Many Parameter System". *Automation and Remote Control*, v. 24, n. 10 (Nov), pp. 1337–1342.

REEVES, C. R., 1993, *Modern Heuristic Techniques for Combinatorial Problems*. Oxford, Blackwell Scientific Press.

RESENDE, M. G. C., RIBEIRO, C. C., 2003, *Greedy randomized adaptive search procedures*. Massachusetts, Kluwer Academic Publishers Group.

RUHE, G., 2003, “Guest Editor's Introduction”. *International Journal of Software Engineering and Knowledge Engineering*, v. 13, n. 5 (Oct), pp. 469-472.

SCHAERF, A., 2002, “Local Search Techniques for Constrained Portfolio Selection Problems”. *Computational Economics*, v. 20, n. 3 (Dec), pp. 177-190.

SIEGEL, S.; CASTELLAN, N. J., 2006, *Estatística não-Paramétrica para Ciências do Comportamento*. 2 ed. São Paulo, Bookman.

SPAIR, R., 2009, *Project Portfolio Optimization*. In: Report C Vision, Level C, Practice Manager.

TOSCANI, L. V.; VELOSO, P. A. S., 2002, *Complexidade de Algoritmos*. Porto Alegre: Instituto de Informática da UFRGS, Editora Sagra Luzzatto.

VERBEETEN, F. H. M., 2006, “Do Organizations Adopt Sophisticated Capital Budgeting Practices to Deal with Uncertainty in the Investment Decision? A Research Note”. *Journal of Management Accounting Research*, v. 17, Issue 1 (Mar), pp. 106-120.

WOHLIN, C., RUNESON, P., HOST, M. et al., 2000, *Experimentation in Software Engineering. An Introduction*, Massachusetts, Kluwer Academic Publishers Group.