UMA ABORDAGEM ECONÔMICA BASEADA EM RISCOS PARA AVALIAÇÃO DE UMA CARTEIRA DE PROJETOS DE SOFTWARE

Hélio R. Costa[1,2], Marcio de O. Barros[2,3], Guilherme H. Travassos[2]

[1]Centro de Computação da Aeronáutica do Rio de Janeiro – Ponta do Galeão s/nº
Ilha do Governador – CEP: 21941-510 – Rio de Janeiro – Brasil

[2] COPPE/UFRJ – Departamento de Ciência da Computação e Engenharia de Sistemas
Caixa Postal: 68511 – CEP: 21945-970 – Rio de Janeiro – Brasil

[3] UNIRIO – Departamento de Ciência Aplicada a Computação
Av. Pasteur 458, Urca – CEP: 22290-240 – Rio de Janeiro – Brasil

heliorcosta@infolink.com.br, {marcio,ght}@cos.ufrj.br

Abstract. *Software engineers have been applying economical concepts to shed light upon the value-related aspects of software development processes. Based on credit risk analysis concepts, we present an approach to estimate the probability distribution of losses and earnings that can be incurred by a software development organization according to its project portfolio. Such approach is built upon an analogy that compares software projects to unhedged loans issued to unreliable borrowers. By applying this approach, an organization may estimate the variability of its expected profits related to a set of software projects. Initial calibrating data were acquired by accomplishing an experimental study.*

Keywords: *Software Engineering Economics, Risk Management, Empirical Studies on Software Engineering.*

Resumo. *Engenheiros de software vêm utilizando teorias econômicas para ressaltar aspectos relacionados à geração de valor relativos ao processo de desenvolvimento de software. Com base em conceitos de análise de riscos de crédito, apresentamos uma abordagem para estimar a distribuição de probabilidade de perdas e ganhos que podem ser auferidos por uma organização, de acordo com sua carteira de projetos. Esta abordagem fundamenta-se em uma analogia que compara projetos de software com empréstimos financeiros. Assim, é possível estimar a variabilidade dos lucros esperados em virtude do desenvolvimento de um conjunto de projetos. Um estudo experimental foi realizado para apoiar a abordagem proposta.*

Palavras Chave: *Economia Aplicada à Engenharia de Software, Gerenciamento de Risco, Estudos Experimentais em Engenharia de Software.*

1. Introdução

A Engenharia de Software é uma atividade econômica como qualquer outra e, como tal, deve ser avaliada não só em relação aos seus problemas técnicos, mas também quanto à

possibilidade de gerar valor para a organização desenvolvedora. Assim, deve haver uma justificativa econômica para a utilização de recursos nos projetos de software. Esta justificativa normalmente é feita em função de seus custos, retornos e riscos envolvidos na escolha das possíveis opções. No desenvolvimento de projetos de software, os custos compõem-se dos gastos com a equipe, hardware, software e infra-estrutura necessária para a execução do projeto. Por sua vez, os retornos são representados pelos lucros esperados ou pelos benefícios proporcionados pelo sistema.

Segundo (Boehm e Sullivan 2000), a Economia aplicada a Engenharia de Software é definida como o campo de estudo que visa fornecer melhorias à Engenharia de Software por meio de raciocínio e justificativas econômicas sobre questões relacionadas a produtos, processos, carteiras de projetos e políticas organizacionais. A noção de que o desenvolvimento de software é uma atividade orientada a valor tem sido introduzida por diversos autores. Por exemplo, Sullivan *et al* (1999) enfatizam a necessidade de pesquisas que desenvolvam abordagens estratégicas de investimentos em Engenharia de Software. Em outro trabalho, Favaro (1998) sugere a utilização de opções reais para avaliar o valor agregado pela reutilização de software.

A preocupação em aproximar as áreas de conhecimento da Engenharia de Software e da Economia é justificada pelos números apresentados em pesquisas de mercado. De acordo com estudos do (Standish Group 2004) cerca de 55 bilhões de dólares foram gastos nos Estados Unidos no ano de 2004 com projetos de software que não tiveram sucesso. Neste cenário, encontramos engenheiros de software cujo principal objetivo é a tomada de decisões técnicas em prol do desenvolvimento de melhores produtos. Por outro lado, temos gerentes e empresários buscando agregar valor a seus investimentos. Boehm e Sullivan (2000) comentam que os engenheiros de software normalmente não estão envolvidos ou não compreendem os aspectos gerenciais de criação de valor para a empresa. Entretanto, gerentes executivos desconhecem os critérios de sucesso para o desenvolvimento de software ou como investimentos em áreas técnicas podem contribuir para a geração de valor.

Porém, quando sistemas de software são desenvolvidos por pequenas empresas, estas podem não resistir a uma conjunção de projetos mal-sucedidos. Assim três questões podem surgir: (i) quanto dinheiro uma organização pode perder ou ganhar em função de um conjunto de projetos de software? (ii) qual a variabilidade (risco) dos lucros ou prejuízos que podem ser obtidos em decorrência de seu desenvolvimento? (iii) que fatores influenciam esta variabilidade? Se estas questões forem respondidas, uma empresa poderá ter melhores condições para controlar sua carteira de projetos de software, visando balancear seus lucros esperados em relação aos riscos incorridos.

Neste artigo, apresentamos uma abordagem para calcular a distribuição de probabilidade de perdas e ganhos de uma organização desenvolvedora de software em função da sua carteira de projetos. A metodologia proposta é baseada na teoria de risco de crédito para operações de empréstimos, a qual é largamente utilizada por instituições financeiras. Dadas algumas premissas, a abordagem proposta permite que uma empresa desenvolvedora de software meça qual a probabilidade de alcançar os lucros esperados a partir de um conjunto de projetos Além disso, a organização pode mensurar a influência de um projeto sobre o risco total de uma carteira e para seu retorno esperado.

Para apoiar esta abordagem, sugerimos uma técnica que permite a avaliação do nível de risco de projetos. Esta avaliação é baseada em uma visão econômica dos elementos que constituem fatores de risco para projetos de software. Um estudo experimental foi planejado e executado para fornecer informações e calibrar a técnica proposta para uma categoria específica de projetos de software. Os resultados obtidos são sumarizados neste artigo.

Além desta introdução, este artigo está estruturado da seguinte forma. Na seção 2 apresentamos o conceito de análise de risco de crédito e propomos uma abordagem para avaliar a probabilidade de perdas e ganhos de uma carteira de projetos de software. Na seção 3 propomos uma técnica para o cálculo de nível de risco de projetos de software de uma determinada categoria. Na seção 4 descrevemos um estudo experimental realizado ao longo deste trabalho. Na seção 5 apresentamos a ferramenta *RISCARE* que foi criada para apoiar as abordagens propostas neste artigo. Finalmente, na seção 6 relatamos algumas limitações e perspectivas futuras deste trabalho.

2. Análise de Risco de Crédito

Para responder às questões descritas na seção anterior, fizemos uma analogia entre o desenvolvimento de sistemas e o mercado financeiro. Nesta analogia, um conjunto de projetos de software é analisado como uma carteira de empréstimos a longo prazo.

Instituições financeiras fornecedoras de empréstimos possuem um especial interesse em determinar a probabilidade de perder ou ganhar uma determinada quantia em dinheiro em função da composição de sua carteira de empréstimos. Esta informação é utilizada para limitar as possíveis perdas financeiras em virtude de eventos de *default*. Um evento de *default* ocorre quando um devedor não consegue cumprir suas obrigações para com a instituição financeira que lhe concedeu um empréstimo.

A análise de riscos de crédito é um conjunto de procedimentos, modelos e técnicas que tem como objetivo determinar, com um determinado nível de certeza, a perda potencial esperada pela manutenção de uma carteira de empréstimos. Ela se baseia na atribuição de uma probabilidade de eventos de *default* para cada devedor. A análise também requer a determinação do retorno esperado em cada empréstimo, uma vez que lucros provenientes de empréstimos bem-sucedidos podem compensar perdas geradas por débitos não saudados.

Projetos de software, assim como empréstimos, possuem probabilidades incertas de sucesso. Um evento de *default* para um projeto de software pode ser associado ao seu cancelamento, independente das causas que levaram a este fato. Projetos de software também possuem um custo, que é o capital empenhado em seu desenvolvimento, como o montante financeiro concedido em um empréstimo. Em certas ocasiões, este capital será perdido caso o projeto falhe, assim como lucros serão obtidos se os projetos alcançarem seus objetivos. Finalmente, projetos bem sucedidos geram lucros para a organização desenvolvedora e podem compensar, até certo nível, perdas causadas pelo insucesso de outros projetos.

Portanto, organizações desenvolvedoras de software podem analisar sua carteira de projetos para determinar a probabilidade de perdas e ganhos financeiros, a fim de manter o negócio economicamente viável mesmo após um (limitado) número de insucessos na execução de projetos de software. Com base nesta analogia,

apresentamos, na próxima seção, uma aplicação dos conceitos de risco de crédito adaptada para o contexto de avaliação de projetos de software.

2.1. Modelagem de Risco de Crédito

As principais abordagens para análise de risco de crédito presentes na literatura financeira são os modelos *Credit Metrics, Credit Risk+* e o KMV (Barbanson *et al* 2004). Cada um desses modelos requer um conjunto distinto de parâmetros e são baseados em premissas particulares.

O modelo KMV analisa a probabilidade de inadimplência de uma empresa em função do valor de mercado de suas ações. O modelo depende da existência de um mercado líquido[1] para as ações da empresa sob análise, cujo valor seria determinado pela negociação entre investidores. No contexto de projetos de software, o valor envolvido na negociação é determinado pelos clientes e pela organização responsável pelo desenvolvimento do projeto. Além disso, os riscos incorridos pelo projeto são geralmente endógenos, ocorrendo por conta de aspectos incertos do processo de desenvolvimento ou do produto. Assim, estes riscos não podem ser determinados por ativos negociados em um mercado aberto[2]. Dadas estas restrições, o modelo KMV não é diretamente aplicável ao problema proposto.

O modelo *Credit Metrics* exige um mercado para negociar empréstimos e ativos de proteção para garantia desses empréstimos, tais como *swaps* de crédito e opções de *default* (Barbanson *et al* 2004). Mais uma vez, como os riscos de um projeto de software são (em sua maioria) endógenos, não existem ativos negociáveis para proteger os projetos de insucessos. Assim, o modelo *Credit Metrics* também não é diretamente aplicável à analogia entre projetos de software e empréstimos.

Por sua vez, o modelo *Credit Risk+* não necessita de um mercado para negociação de ativos de proteção e a única informação necessária em relação a um devedor é a sua probabilidade de *default* para cada empréstimo. Se dispusermos de um número que represente a probabilidade de insucesso de um projeto de software, a aplicação do modelo *Credit Risk+* para análise de carteira de projetos de software passa a ser viável. No restante desta seção, assumiremos a existência deste número. Na seção 3, apresentaremos uma técnica que permite estimar este número.

O modelo *Credit Risk+* é composto por um conjunto de três modelos distintos, cada um construído sobre o modelo precedente e adicionando maiores detalhes ao seu processo de cálculo. O primeiro modelo é baseado em uma fórmula analítica que calcula a distribuição de probabilidade de perdas e ganhos de uma carteira de empréstimos. Cada empréstimo é caracterizado por uma quantidade de dinheiro envolvida, o retorno esperado e o risco inerente desta negociação. O modelo assume que todo o dinheiro emprestado é perdido em caso de um evento de *default*. Além disto, os devedores são normalmente agrupados de acordo com a sua habilidade em cumprir com suas obrigações, sendo que para cada grupo é atribuída uma probabilidade de *default*.

---

[1] Um mercado líquido é aquele onde as ações de uma ou mais empresas estão suficientemente distribuídas entre investidores de tal forma que nenhum investidor em particular pode influenciar no valor das ações.

[2] Um mercado aberto é aquele onde qualquer pessoa ou empresa pode comprar ou vender ativos.

Esta classificação em grupos é possível no mercado financeiro devido à grande quantidade de devedores e dados históricos sobre empréstimos anteriores. No contexto de projetos de software, a aplicabilidade deste agrupamento se reduz, dado que uma empresa geralmente não possui tantos projetos em desenvolvimento simultâneo, a fim de que possam ser classificados e uma probabilidade de insucesso seja atribuída para todo o grupo. O segundo e o terceiro modelos do *Credit Risk+* também levam em consideração a variabilidade da probabilidade de default e a distribuição de devedores entre setores econômicos. Neste trabalho apenas o primeiro modelo está sendo considerado.

A proposta para projetos de software apresentada neste artigo é baseada no primeiro modelo do *Credit Risk+*, que lida com probabilidades de *default* constantes, ou seja, não leva em consideração a variabilidade destas probabilidades. Mapeando este modelo para o contexto de projetos de software, a abordagem proposta assume que cada projeto tem uma probabilidade de insucesso fixa.

Com o objetivo de estabelecer o capital necessário para manter uma carteira de empréstimos, o *Credit Risk+* utiliza a distribuição de probabilidade Poisson (Johnson e Bhattacharyya 1977) para descrever o número de eventos de *default* que podem ocorrer em uma carteira de empréstimos dado o número de devedores de cada grupo. De acordo com o número de *defaults* em cada grupo, os lucros esperados e a quantidade de dinheiro investida em cada empréstimo, o modelo estima o retorno esperado para a carteira e sua variabilidade em relação a cada grupo.

Como as organizações desenvolvedoras de software geralmente não possuem um número elevado de projetos concorrentes em sua carteira, é pouco prático formar grupos de projetos. Por conta desta característica, ao invés de determinar a probabilidade de *default* para cada grupo e utilizar a distribuição de Poisson, calculamos a probabilidade de *default* para cada projeto e utilizamos a distribuição de Bernoulli (Johnson e Bhattacharyya 1977) para descrever seu comportamento em termos de sucesso ou insucesso.

Ao substituir a função de distribuição de probabilidade original, a fórmula analítica do modelo *Credit Risk+* deixa de ser aplicável. Para efetuar os cálculos do modelo, sugerimos o uso da Simulação de Monte Carlo para estimar a distribuição de probabilidade de perdas e ganhos de uma carteira de projetos de software, de acordo com a proposta de (Aragão *et al* 2003).

O modelo *Credit Risk+* assume algumas premissas que precisam ser consideradas para estabelecer os limites de nossa abordagem. Por exemplo, o modelo presume que todos os custos são incorridos quando o empréstimo é fornecido e os retornos são recebidos quando os empréstimos são pagos. Esta premissa pode não ser verdadeira para todos os projetos de software, pois alguns possuem previsões de fluxo de caixa (positivos e negativos) ocorrendo ao longo do ciclo de vida do projeto. A Tabela 1 apresenta uma comparação entre as premissas assumidas para o contexto do mercado financeiro e de projetos de software.

Tabela 1. Premissas financeiras e de software

Mercado Financeiro	*Projetos de Software*
Cada empréstimo possui seu próprio risco	Cada projeto possui seu próprio risco
Estados do devedor: inadimplente ou não	Estados do projeto: cancelado ou não
O valor envolvido na operação de empréstimo é acordado antes da operação ser realizada	Os custos de projeto são definidos antes de sua realização
O pagamento do empréstimo é realizado ao fim do período acordado	O pagamento do projeto é feito após sua conclusão
Em caso de evento *default* as perdas possuem um valor fixo (o valor acordado para o empréstimo), desconsiderando-se a taxa de retorno prevista	Em caso de insucesso do projeto, a perda possui um valor fixo (os custos do projeto), independente do retorno esperado
Não existe correlação entre os eventos *default,* exceto por fatores externos que podem afetar os devedores de maneira similar	Não existe correlação entre os insucessos dos projetos, exceto por riscos externos (sistêmicos) que podem afetar os projetos de maneira similar

Assumindo estas premissas e por meio das simulações, é possível estabelecer a distribuição de probabilidade de perdas e ganhos para qualquer número de projetos de uma carteira de uma organização, baseado apenas na probabilidade de insucesso dos projetos (nível de risco), nos seus custos de desenvolvimento e nos retornos esperados. O cálculo para a distribuição de probabilidade ocorre da seguinte forma:

- O simulador gera um valor aleatório para cada projeto envolvido na simulação segundo a distribuição de Bernouille. Estes números representam a possibilidade de sucesso ou fracasso do projeto da simulação corrente;
- Caso o projeto seja definido como um insucesso, o resultado da operação é a perda dos custos determinados para este projeto;
- Se o projeto for bem-sucedido, obterá um lucro e o resultado da operação é calculado pela diferença entre os retornos esperados e os custos do projeto;
- Ao final de cada simulação, os resultados das avaliações dos projetos (custos ou lucros) são somados. Este resultado representa a quantidade de dinheiro perdida ou ganha pela empresa no desenvolvimento dos referidos projetos;
- A simulação é repetida diversas vezes a fim de fornecer uma melhor estimativa dos retornos esperados para a carteira de projetos. Para garantir a convergência do processo de cálculo, sugere-se que as simulações sejam repetidas por, pelo menos, 10.000 ciclos;
- Ao final de todas as simulações, as perdas ou ganhos são agrupados e mapeados para o domínio da freqüência. Neste mapeamento, o espaço entre a pior perda e o melhor ganho é dividido em um número previamente determinado de intervalos de uma mesma dimensão, realizando-se uma contagem do número de perdas e ganhos gerados pela simulação em cada intervalo;

- Finalmente, a percentagem de perdas e ganhos em cada grupo é calculada em relação ao número total de simulações realizadas e estes valores são plotados em um gráfico de barras de maneira cumulativa.

A fim de ilustrar a abordagem proposta, tomaremos como exemplo uma empresa fictícia com uma carteira de projetos descrita pela Tabela 2, onde cada projeto possui o custo planejado para seu desenvolvimento e o retorno financeiro esperado[3]. Para que o cálculo seja efetuado, faz-se necessário também os níveis de risco para os projetos (probabilidades de *default*), que representam suas chances de insucesso. Estes valores estão sendo assumidos apenas para efeito do exemplo. No entanto, serão mais detalhadamente elucidados na seção 3.

Tabela 2. Projetos com níveis de risco, custos e retornos

Projeto	*Nível de Risco*	*Custo*	*Retorno*
#1	10%	$ 10 K	$ 25 K
#2	20%	$ 20 K	$ 35 K
#3	15%	$ 30 K	$ 45 K
#4	25%	$ 40 K	$ 50 K
#5	20%	$ 50 K	$ 75 K

Após aplicar a simulação de Monte Carlo 10.000 vezes com os parâmetros fornecidos no exemplo, a distribuição de probabilidade cumulativa de perdas e ganhos que podem ser obtidas pela empresa está representada na Figura 1.

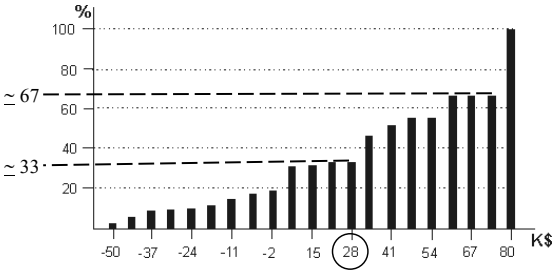

Figura 1. Distribuição de probabilidade

Os resultados expressos na Figura 1 são apresentados de forma cumulativa, ou seja, as barras verticais representam a soma da probabilidade da quantia anterior mais a probabilidade do valor sob observação. Desta forma, pelos resultados obtidos, podemos observar que a probabilidade de se perder dinheiro nesta carteira de projetos é cerca de 19% (representada pelas probabilidades cumulativas dos valores negativos). Podemos observar, também, que a probabilidade de se ganhar $28K ou mais é aproximadamente 67%, enquanto que a chance de obter $80K (lucro máximo possível) é cerca de 36% (100% - 67%, que é a probabilidade cumulativa de se obter valores menores que 80K).

[3] Assume-se que tanto o custo quanto o retorno esperado foram calculados por algum método quantitativo apropriado para o projeto.

Diversas simulações podem ser realizadas mudando-se os parâmetros utilizados nos cálculos (custo, nível de risco e retorno), com o objetivo de observar qual a melhor combinação possível para a carteira de projetos e a aplicação de recursos, por exemplo:

- Diminuir o nível de risco e diminuir o custo;
- Tentar aumentar o retorno ou diminuir o nível de risco;
- Apenas diminuir os custos;
- Reduzir o risco, aumentando o custo, utilizando um plano de contingência;
- Apenas tentar aumentar o retorno;
- Eliminar um ou mais projetos da carteira.

Como exemplo de variação de parâmetros, temos a situação onde, após uma análise das possibilidades a serem feitas na carteira de projetos, gerentes ou executivos da organização resolvem alterar parâmetros conforme pode ser observado na Tabela 3.

Tabela 3. Projetos com níveis de risco, custos e retornos alterados

Projeto	*Nível de Risco*	*Custo*	*Retorno*
#1	10%	$ 10 K	$ 30 K
#2	18%	$ 22 K	$ 35 K
#3	15%	$ 25 K	$ 45 K
#4	20%	$ 45 K	$ 60 K
#5	15%	$ 55 K	$ 80 K

Realizando mais uma vez a simulação, podemos obter uma distribuição semelhante à apresentada na Figura 2.

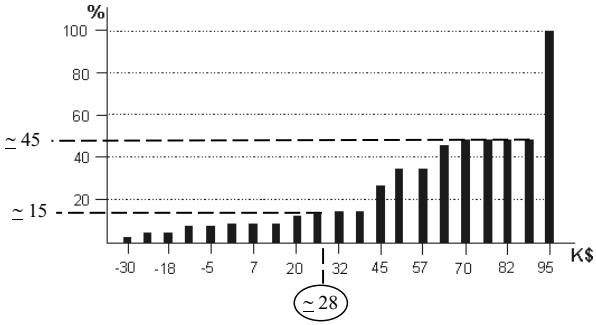

Figura 2. Distribuição de probabilidade alterada

Nesta situação, a probabilidade de se ter prejuízo (valores negativos) com a carteira de projetos é cerca de apenas 6%, ao passo que na situação anterior era de 19%. A chance de se ganhar cerca de $28K ou mais foi aumentada para 85% e o lucro máximo possível passou para $95K, sendo a probabilidade de se obter este valor de aproximadamente 55%.

Na próxima seção apresentamos uma técnica para calcular a probabilidade de insucesso de um projeto de software, informação necessária para a realização dos cálculos da abordagem apresentada nesta seção.

3. Quantificação de Riscos

Riscos de projetos de software podem ser classificados em dois grupos: riscos sistêmicos e riscos específicos. Os riscos sistêmicos são aqueles que afetam todos os projetos de software de uma determinada categoria. Como exemplos de categorias de projetos de software, podemos citar os sistemas de informação, os sistemas militares, os sistemas comerciais, entre outros. Qualquer projeto em uma categoria está sujeito, até certo nível, aos riscos sistêmicos da categoria. Riscos específicos, por sua vez, são os fatores de risco associados às características específicas de um projeto.

Esta separação entre riscos sistêmicos e específicos tem suas origens na Economia. Os riscos sistêmicos (no mercado financeiro) são oriundos das condições econômicas do país onde a empresa está localizada, de políticas governamentais, do cenário econômico internacional e de outros fatores globais que afetam todas as empresas que atuam em um mesmo mercado. Riscos específicos são relacionados a uma determinada organização, condições locais de mercado, administração, reputação da empresa e diversos outros fatores.

Fundamentados nestes conceitos, foi criada uma técnica para avaliar o nível de risco de projetos, de forma a viabilizar a abordagem proposta na seção 2. Esta técnica baseia-se num questionário construído a partir da taxonomia de riscos apresentada por (Carr *et al* 1993), que foi complementada com outros questionários e taxonomias apresentadas na literatura (Boehm 1991; Barki 1993; Jones 1994; Karolak 1996; Moyniham 1997 e Wallace 1999).

Após a análise dos questionários, as fontes de risco encontradas foram reagrupadas por semelhança. Algumas questões foram eliminadas pelo fato de estarem presentes em mais de um questionário. Outras questões necessitaram ser reescritas para que pudessem ser enquadradas no modelo proposto para este trabalho. Assim, todas as perguntas foram expressas de forma que quanto mais presente em um projeto, mais esta fonte de risco contribui para o seu insucesso. Vale ressaltar que o sentido original das perguntas não foi alterado quando tiveram que ser reescritas.

O questionário possui 211 questões, que são classificadas em dez grupos, denominados fatores de risco. Cada questão pertence a um único fator, tendo como objetivo descrever de forma detalhada o conceito abstrato representado pelo fator de risco, relacionando-o com elementos mais práticos que podem ser avaliados pelo gerente, de acordo com as características do projeto em análise. Os fatores de risco determinados são: Análise, Projeto, Codificação, Testes, Planejamento, Controle, Equipe, Política/Estrutura Organizacional, Contratos e Clientes. Maiores detalhes podem ser encontrados em http://www.cos.ufrj.br/~heliorc/riskquest.html.

Neste trabalho, utilizamos o conceito de risco sistêmico para mapear as diferenças de importância entre os fatores de risco. Reconhecemos que podem existir diferenças entre estes fatores em diversos projetos de software, mas utilizamos a premissa de que, dentro de uma mesma categoria de sistemas, esta diferença pode ser

explicada pelos riscos específicos, permanecendo fixas as relações entre os fatores de risco sistêmicos.

Para se determinar o nível de risco de um projeto, o primeiro passo consiste em responder ao questionário de identificação de risco. Existem sete respostas possíveis para cada questão: o valor 0 (zero) indica que o problema tratado pela questão não representa um risco para o projeto. Os demais valores (1 a 5) são codificados de uma escala de ordenação subjetiva, que varia de Risco Muito Baixo (1) a Risco Muito Alto (5). Esta operação de codificação é baseada nos estudos de (Likert 1932), onde números podem ser utilizados para expressar valores de uma escala ordinal. O gerente também possui a opção de assinalar a uma questão como não relevante para o projeto (NR).

No segundo passo da técnica de quantificação de risco, o gerente calcula a mediana dos valores atribuídos às respostas de cada fator de risco. Questões marcadas como não relevantes (NR) são tratadas como valores ausentes e não são consideradas no cálculo das medianas.

No terceiro passo, divide-se a mediana calculada de cada fator por cinco (5) e multiplica-se o resultado pelo valor de ajuste do fator. Dado que o valor máximo de cada mediana é cinco (5), a divisão visa normalizar estes valores, transformando-os em um número compreendido no intervalo entre 0 e 1. A multiplicação pelo valor de ajuste tem o objetivo de ponderar este fator em relação a sua importância sistêmica em relação aos outros fatores. Finalmente, no quarto passo, os números resultantes referentes a cada fator são somados, combinando em um único número os riscos sistêmicos e específicos do projeto, representando seu nível de risco.

A Tabela 4 apresenta um exemplo do cálculo do nível de risco de um projeto. Neste exemplo, apenas dois fatores de risco são considerados. O primeiro possui três (3) questões e um valor de ajuste de 70%. O segundo fator possui duas (2) questões e um valor de ajuste de 30%.

Tabela 4. Cálculo do nível de risco

	Fator 1			***Fator 2***	
Responda o questionário	Q1	Q2	Q3	Q1	Q2
	2	4	3	2	4
Calcule a mediana das questões	2, 4 e 3 = 3			2 e 4 = 3	
Ajuste os valores das medianas	$\frac{3}{5} * 70\% = 42\%$			$\frac{3}{5} * 30\% = 18\%$	
Some os valores ajustados	Nível de risco = 60%				

O valor de ajuste captura a relevância sistêmica de um fator de risco para uma categoria específica de sistemas. Algumas propriedades dos valores de ajuste devem ser ressaltadas: (*i*) a soma de todos os valores de ajuste deverá ser sempre 100%; e (*ii*) quanto maior a relevância de um fator, maior será o seu valor de ajuste. Para calibrar a técnica aqui proposta, realizamos um estudo experimental para identificar estes valores. A próxima seção apresenta o planejamento do estudo e os resultados obtidos em sua primeira execução.

4. Estudo Experimental

O agrupamento das categorias para o cálculo do nível de risco do projeto exige que seja atribuído um peso para cada fator (valor de ajuste), permitindo assim que os resultados obtidos nas respostas a suas perguntas possam ser agregados com os resultados obtidos nas respostas às perguntas que compõem as outras categorias. No entanto, esses valores não são conhecidos, surgindo, então, a necessidade de se realizar um estudo experimental para identificá-los. A estratégia adotada neste trabalho foi a Pesquisa de Opinião.

Devido aos diferentes tipos de projetos de software que podem ser desenvolvidos para um número cada vez maior de domínios, é suposto que os valores de ajuste, requeridos pela técnica de avaliação de risco apresentada na seção anterior, podem variar em função das diferentes categorias de sistema. Com o objetivo de reduzir o escopo desta pesquisa e aumentar sua precisão, limitamos o estudo experimental à avaliação dos valores de ajuste para Sistemas de Informação.

Objetivo: o ponderação dos fatores de risco foi realizada no sentido de determinar o grau de criticalidade no contexto de um projeto de software. Entende-se por criticalidade o grau de importância com que um fator contribui para o **insucesso** de um projeto, ou seja, quanto mais crítico um fator, maior será a probabilidade de **insucesso** em um projeto, caso ocorra algum problema nessa área.

Participantes: durante a execução do estudo, 50 especialistas foram entrevistados e seus perfis foram computados para uma melhor caracterização e análise dos dados. Todos os participantes concordaram em participar voluntariamente no estudo e assinaram um formulário de consentimento desta participação. Dos 50 participantes, sete (7) possuíam o grau de Doutor em Ciência da Computação, quinze (15) eram Mestres, dezenove (19) possuíam algum tipo de especialização e nove (9) eram apenas graduados. Dentre os participantes, onze (11) eram pesquisadores acadêmicos ou professores, enquanto que trinta e nove (39) eram profissionais de empresas de desenvolvimento de software da cidade do Rio de Janeiro. Durante o estudo, vinte e seis (26) diferentes instituições foram visitadas. Trinta e quatro (34) participantes atuavam, no momento da entrevista, como gerentes de projeto, ao passo que dezesseis (16) eram analistas sênior. A média de experiência em desenvolvimento de software de cada participante foi de doze (12) anos, enquanto que, em média, participaram de quatorze (14) projetos.

Seleção dos Participantes: O método adotado para a escolha dos participantes foi a *Amostragem* de *Quota*, onde foram selecionados vários elementos da população alvo, que são os gerentes de projeto, estudantes, professores e profissionais de empresas desenvolvedoras de software. Para cada um desses elementos a escolha foi feita com *Amostragem por Conveniência*. Desta forma, os participantes selecionados representam uma amostra de todos os elementos da população, mas não foram escolhidos de forma aleatória.

Caracterização dos Projetos: Foi solicitado aos participantes, que escolhessem como parâmetro de sua avaliação o tamanho de projeto que mais representava sua experiência. Os projetos escolhidos foram caracterizados pelo esforço requerido para seu desenvolvimento (medidos em homem/mês). Os tamanhos de projetos foram usados na análise dos dados para agrupar a opinião dos participantes, de forma que

valores de ajuste diferentes pudessem ser determinados para tamanhos distintos de projetos (pequenos, médios e grandes).

Grupamento de Participantes e Projetos: Os participantes foram classificados, de acordo com sua experiência, em três grupos: pouca, média e grande experiência. Um formulário de caracterização capturou a experiência acadêmica e industrial do participante. Exemplos destas informações incluem: número de anos trabalhando em desenvolvimento de sistemas, quantidade de projetos desenvolvidos, formação acadêmica e grau de experiência em gerência de risco. Estas informações nos permitiram sumarizar a experiência dos participantes em um único número (que variava de 1 a 15, sendo o maior representando a maior experiência) e agrupar os participantes usando a abordagem proposta por (Farias *et al* 2003).

Após este grupamento, foi possível determinar um peso comum para a opinião de participantes de um mesmo grupo. Determinou-se um peso de 1 (um) para participantes co pouca experiência. O peso dos outros grupos foram determinados pela divisão de seus pesos médios pelo valor do peso médio original calculado para o grupo de pouca experiência. De acordo com esta classificação, a Tabela 5 expressa o número de participantes, as medias e os pesos finais calculados para cada grupo.

Tabela 5. Participantes e seus pesos

	Pouca Exeperiência	***Média Exeperiência***	***Grande Exeperiência***
# Participantes	13	28	9
Peso Médio	4.09	6.62	10.78
Peso Final	1.0	1.61	2.63

Instrumentação: A ferramenta escolhida para a realização desse estudo foi o software MACBETH desenvolvido por (Costa e Vasnick 1995). Esta ferramenta foi desenvolvida especialmente para permitir a ordenação de fatores em um processo decisório baseado em critérios pré-definidos, onde através da comparação aos pares de cada fator envolvido e o julgamento de valor dessa comparação chega-se à ordenação desejada.

Durante o estudo foi utilizada uma versão *demo* da ferramenta. No entanto, apesar desta limitação, MACBETH possui, nesta versão, todas as funcionalidades necessárias para a realização do estudo e coleta dos dados.

Eliminação de *Outliers*: Após a ponderação dos votos dos participantes, os dados foram analisados e dados discrepantes foram eliminados a fim de não distorcer os resultados. O critério de eliminação foi determinado pelo cálculo da média e do desvio padrão em relação aos votos dos 10 fatores de risco. Desta forma, todos os valores que se encontravam a uma distância igual ou superior a dois (02) desvios padrão em relação à média dos valores foram eliminados. Dos 500 dados brutos obtidos durante o estudo, apenas 14 foram rejeitados pelo critério de eliminação de *outliers*.

Análise Estatística: Os dados considerados válidos para o estudo foram submetidos a um Teste-T com nível de significância de 0,05 (5%) para verificar se existia diferença estatística entre os valores obtidos e se os resultados podiam ou não ser generalizados. Foram comparadas as médias dos valores de ajuste dos diferentes fatores de risco em relação aos tamanhos dos projetos e a influência da experiência dos participantes em relação aos votos.

As comparações realizadas demonstraram não haver nenhuma diferença significativa para os fatores *Análise*, *Projeto*, *Codificação*, *Teste*, *Planejamento* e *Contratos*. Os demais fatores – *Equipe*, *Controle*, *Política/Estrutura* e *Clientes* – apresentaram uma pequena, mas significante, diferença entre os valores obtidos para projetos pequenos e grandes, mas nenhuma diferença entre projetos pequenos e médios ou médios e grandes.

Independente da grande semelhança dos resultados estatísticos obtidos decidiu-se manter, para utilização na técnica de avaliação de riscos de projetos proposta, todos os valores calculados neste estudo, pelo fato destes resultados representarem apenas uma primeira abordagem para este tipo de informação em projetos de software.

Valores de Ajuste dos Fatores de Risco: Após a ponderação dos votos de todos os participantes, eliminação de *outliers* e separação por tamanho de projeto, as médias dos resultados foram normalizadas a fim de respeitar a regra de que a soma de todos os pesos dos fatores para um projeto de um dado tamanho deve somar 100%.

Os resultados para os valores de ajuste para cada fator e em seus diferentes tamanhos de projeto estão expressos na Tabela 6 e representam o grau de importância em um projeto de desenvolvimento de software para Sistemas de Informação.

Tabela 5. Valores de ajuste

	Pequeno (%)	***Médio (%)***	***Grande (%)***
Análise	12,36	12,57	10,78
Projeto	8,59	7,52	6,23
Codificação	4,84	4,13	4,00
Teste	7,36	6,17	5,82
Planejamento	15,26	13,04	13,85
Controle	10,39	11,64	12,19
Equipe	11,28	11,53	12,63
Contratos	3,40	5,37	4,60
Política/Estrutura	11,41	12,47	14,11
Clientes	15,11	15,57	15,79

Maiores detalhes sobre o estudo experimental podem ser encontrados em (Costa 2004).

5. A Ferramenta *RISICARE*

Em virtude da grande quantidade de questões que precisam ser respondidas, das operações necessárias para o cálculo do nível de risco dos projetos e das simulações que geram a distribuição de probabilidade de perdas e ganhos de uma carteira de projetos, torna-se impraticável para um gerente realizar estas atividades manualmente. Portanto, buscando apoiar estas atividades, a ferramenta *RISICARE* foi concebida e implementada, de tal forma que possui cinco módulos:

- **Caracterização de Projetos:** utilizado para fornecer dados tais como: Nome do Projeto, Tamanho da Equipe¸ Duração¸ Custo e Retorno;
- **Questionário:** dividido em dois submódulos (Edição e Respostas). Em Edição pode-se cadastrar novas questões, alterar as existentes e personalizar o questionário para um projeto ou uma organização. Em Respostas o gerente responde as questões, de acordo com as opções descritas na seção 3;

- **Cálculo do Nível de Risco:** utilizado para calcular o nível de risco dos projetos;
- **Listagem de Projetos:** permite que o usuário tenha acesso a todos os projetos cadastrados no sistema, apresentando informações sobre o seu *status* em relação ao preenchimento do questionário, o custo de desenvolvimento, o retorno previsto e o nível de risco, caso tenha sido calculado. Três possíveis operações podem ser realizadas neste módulo: (*i*) excluir os projetos, (*ii*) calcular o nível de risco de todos os projetos listados e (*iii*) transportar os projetos para a área de simulação;
- **Simulação:** pode-se realizar simulações com os projetos selecionados, salvá-las para futuras análises e registrar comentários sobre as simulações.

As Figuras 3 e 4 ilustram respectivamente um exemplo das telas de respostas do questionário e de simulação.

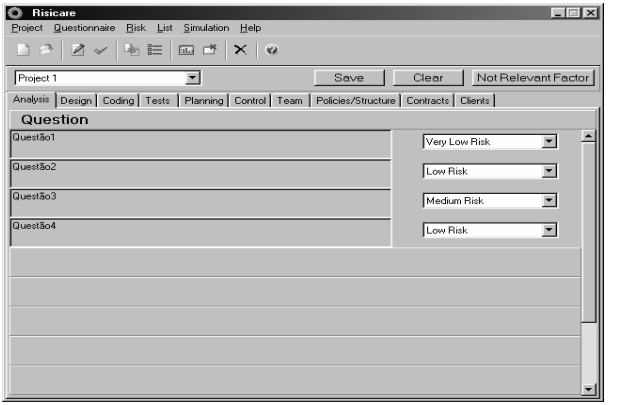

Figura 3. Tela de questionário

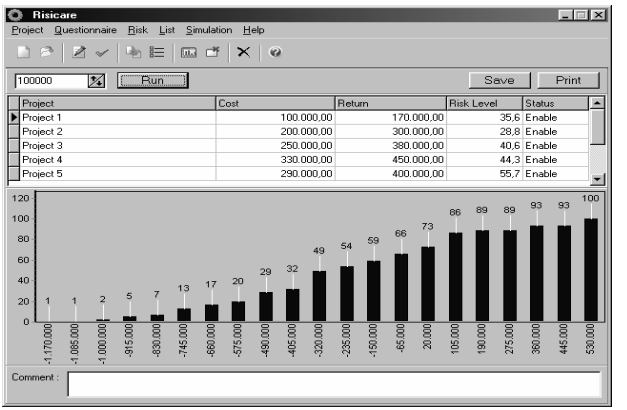

Figura 4. Tela de simulação

6. Limitações e Perspectivas Futuras

Reconhecemos algumas limitações em nossa abordagem. A primeira é o fato de que, para aumentar sua confiabilidade, o estudo experimental foi restrito a Sistemas de Informações. Para que os resultados do cálculo de nível de risco possam ser generalizados, pretendemos, em um futuro próximo, repetir o estudo para outras categorias de sistemas.

A segunda limitação refere-se às restrições impostas pelo modelo *Credit Risk+*, onde a dimensão do tempo e a impossibilidade de analisar fluxos de caixa ao longo do projeto não podem ser considerados. Pretendemos expandir nossa pesquisa no sentido de possibilitar aplicar um modelo que suporte estruturas mais complexas de fluxo de caixa.

Como possíveis evoluções da ferramenta *RISICARE*, seria interessante agregar conhecimento e inteligência à ferramenta, no sentido de auxiliar gerentes de projeto na análise dos resultados obtidos nas simulações, bem como permitir que os gerentes possam armazenar os conhecimentos adquiridos para consulta em novas avaliações.

Agradecimentos

Gostaríamos de agradecer a todas as pessoas que participaram do estudo experimental realizado, pois sem suas contribuições esse trabalho não seria possível. Agradecemos também à Força Aérea Brasileira e, especialmente, ao CNPq pelo apoio prestado ao prof. Guilherme H. Travassos.

Referências

Sullivan K., Chalasani P., Jha S., Sazawal V. (1999) "Software Design as an Investment Activity: A Real Options Perspective" in Real Options and Business Strategy: Application to Decision Making, L. Trigeorgis, Consulting Editor, Risk Books.

Favaro J.M., Favaro K.R and Favaro P.F. (1998) "Value Based Software Reuse Investment, Annals of Software Engineering 5, pp. 5 – 52.

Boehm B.W., Sullivan K. (2000) "Software Economics: A Roadmap, in The Future of Software Engineering", 22nd International Conference on Software Engineering.

Standish Group (2004) "Chaos Demographics – 2004 Third Quarter Research Report" The Standish Group International Inc.

Barbanson R. et al. (2004) "An Introductin to Credit Risk with a Link to Insurance" AFIR, Working Grioup.

Johnson R., Bhattacharyya G. (1977) "Statistical Concepts and Methods" John Wiley & Sons. New York.

Aragão C. S. L., Carvalho L. E. Z. L., Barros M. O. (2003) "Análise do risco de uma carteira de crédito por meio de simulação de Monte Carlo" Resenha BM&F nº 152.

Carr M. J., Konda S.L, Monarch I., Ulrich F.C., Walker C.F. (1993) Taxonomy-Based Risk Identification" Technical Report CMU/SEI–93-TR-6, Software Engineering Institute, Carnegie Mellon University, EUA, July.

Boehm B.W. (1991) "Software Risk Management: Principles and Practices" IEEE Software, vol. 8, n. 1, January, pp. 32-41.

Barki H., Rivard S., Talbot J. (1995) "Toward an Assessment of Software Development Risk" Journal of Management Information Systems, v. 10, n. 2, p. 203-225.

Jones C. (1994) "Assessment and Control of Software Risks" Yourdon Press Computing Series.New Jersey.

Karolak D.W.(1996) "Software Engineering Risk Management" Los Alamitos, CA: IEEE, Computer Society Press.

Moynihan T.(1997) "How Experienced Project Managers Assess Risk" IEEE Software, v. 14, n. 3, p. 35-41, May/June.

Wallace L. (1999) "The development of an instrument to measure software project risk" PhD Thesis - College of Business Administration, Georgia State University, Georgia.

Likert R. (1932) "A technique for the Measurement of Attitudes" Arquives of Psycology.

Farias L.L,, Travassos G.H., Rocha A.R.C. (2003) "Managing Organizational Risk Knowledge" Journal of Universal Computer Science, Vol. 9, No. 7, pp. 670-681.

Miranda E. (1999) "Establishing Software Size Using the Paired Comparisons Method" Ericsson Research, Canada.

Bana e Costa C.A., Vansnick J.C. (1995) "A theoretical Framework for Measuring Attractiveness by a Categorical Based Evaluation Technique (MACBETH)" In: Clímaco, J., Multicriteria Analysis. Berlin: Springer.

Costa H.R., Barros M.O., Travassos G.H.(2004) "Software Project Risk Evaluation Based on Specific and Systemic Risks" in: *Proceedings of the 16th International Conference of Software Engineering and Knowledge Engineering*.