

Avaliação *Fuzzy* da Qualidade de Produtos de *Software* numa Fundação Pública Estadual

Alfredo N. P. Boente
Instituto Superior de
Tecnologia em Ciência da
Computação do Rio de Janeiro
(IST-Rio)

Universidade Estácio de Sá
(UNESA)

alfredoboente@faetec.rj.gov.br

Jesús Domech Moré
Universidade Estácio de Sá
(UNESA)

jesus.more@estacio.br

Harvey José S. R. Cosenza
Universidade Estácio de Sá
(UNESA)

harveycosenza@gmail.com

Abstract. *This paper discusses the use of the theory of fuzzy sets as a tool for evaluating the quality of software products developed in a statewide public foundation. The research was descriptive character and achievement in the foundation board computer. Structured questionnaires were used as a tool for analysis of criteria for quality of software products, and application of Microsoft, Excel for tabulation, data processing and creation of graphics. Based on the results, were presented the degree of presence, the degree of importance and the index of quality of software products developed. Finally, it is expected with this article, identify vulnerabilities that may result in software products with a quality score lower than expected by project managers.*

Resumo. *Este artigo trata do uso da teoria dos conjuntos fuzzy como ferramenta para a avaliação da qualidade de produtos de software desenvolvidos numa fundação pública estadual. A pesquisa teve caráter descritivo e realização na diretoria de informática dessa fundação. Foram utilizados questionários estruturados como instrumento de análise dos critérios de qualidade de produtos de software; e o aplicativo da Microsoft, o Excel, para tabulação, processamento dos dados e criação de gráficos. Com base nos resultados obtidos, foram apresentados o grau de presença, o grau de importância e o índice de qualidade de produtos de software desenvolvidos. Por fim, espera-se com esse artigo, identificar pontos vulneráveis que possam resultar em produtos de software com um índice de qualidade inferior daquele esperado pelos gerentes de projeto.*

1. Introdução

Na área de produção de *software* há uma urgente necessidade de focar a questão de qualidade. Segundo Mecenas e Oliveira (2005), a produção de *software* deixou de ser, há algum tempo, uma atividade baseada apenas na intuição ou na experiência dos desenvolvedores. O processo de desenvolvimento de *software* tem sido objeto de inúmeros estudos, há mais de três décadas, numa tentativa de derivar modelos que

possibilitem o gerenciamento das fases de produção e assegurem que os produtos tenham a qualidade desejada pelos consumidores.

Qualidade é o sucesso para o negócio de *software*, como em qualquer outro. Uma questão-chave é verificar se realmente os produtos de *software* são produzidos com um índice de qualidade satisfatório. Normas de engenharia de *software* garantem que produtos de *software* sejam desenvolvidos com qualidade. Soares (2004) propõe que, ao se aplicarem as normas de engenharia de *software*, os resultados iniciais em termos de qualidade, de confiança, de datas de entrega e de custo sejam promissores.

Tendo em vista que a avaliação da qualidade de *software* é um aspecto de representação imprecisa, composto, em sua maioria, por conceitos subjetivos e de avaliação não trivial, utiliza-se a teoria dos conjuntos *fuzzy* como ferramenta para avaliação do índice de qualidade de produtos de *software*, permitindo analisar e interpretar dados que possam ser obtidos a partir da opinião desses gerentes de projetos envolvidos nesse processo de desenvolvimento.

2. Qualidade de *Software*

Rezende (1999, p. 97) afirma que “um *software* ou sistema de informação tem qualidade quando está adequado à empresa, ao cliente e/ou usuário e atende a padrões de qualidade predefinidos”.

Pressman (2006, p. 724) define qualidade de *software* como:

A conformidade a requisitos funcionais e de desempenho explicitamente declarados, a padrões de desenvolvimento claramente documentados e a características implícitas que são esperadas de todo *software* profissionalmente desenvolvido.

Poucas organizações podem escapar da introdução de projetos em suas atividades diárias. Este fenômeno tem resultado em um maior interesse na aplicação da gestão de projetos como uma metodologia formal de gestão da qualidade (MORRISON e BROWN, 2004 apud PAPADIUK e SANTOS, 2008, p. 3). A gestão da qualidade faz parte das atividades que envolvem o gerenciamento e controle de projetos. O controle da qualidade do projeto inclui os processos requeridos para garantir que o projeto de *software* irá satisfazer ou até superar as expectativas de seu contratante (BOENTE, 2003, p. 66).

3. Qualidade de Produtos de *Software*

A qualidade do produto é o conjunto das propriedades que determinam sua habilidade em satisfazer as necessidades para as quais ele foi criado (JURAN, 1991).

Rocha (1994) diz que qualidade pode ser entendida como um conjunto de características a serem satisfeitas em um determinado grau, de modo que o produto de *software* atenda às necessidades explícitas e implícitas de seus usuários. A qualidade deve ser construída ao longo do processo do desenvolvimento do *software* e também após a entrega do mesmo. De acordo com Belchior (1997, p. 15) para que se obtenha a qualidade desejada de produtos de *software*, fazem-se necessários modelos que viabilizem a avaliação da qualidade desses produtos. As organizações internacionais de normalização ISO/IEC vêm trabalhando em conjuntamente em um modelo que permita avaliar a qualidade dos produtos de *software* (WEBER e ROCHA, 1999).

De acordo com Tsukumo (1997, p. 16):

A qualidade de um produto de *software* é resultante das atividades realizadas dos processos de desenvolvimento do mesmo. Avaliar a qualidade de produtos de *software* é verificar, através de técnicas e atividades operacionais o quanto os requisitos são atendidos. Tais requisitos, de uma maneira geral, são a expressão das necessidades, explicitados em termos de quantitativos ou qualitativos, e tem por objetivo definir as características de um *software* [...].

Pode-se afirmar que a garantia da qualidade do *software* está diretamente relacionada com a garantia do processo e com a garantia do produto de *software* a ser confeccionado e, esta garantia envolve aplicação de métodos técnicos, realização de revisões técnicas formais, atividades de teste de *software*, aplicação de padrões, controle de mudanças, métrica de *software* e manutenibilidade do produto (BOENTE, OLIVEIRA e ALVES, 2008, p.8). Numa perspectiva de mensuração, qualidade de *software* deve ser definida em termos de atributos de produtos de *software* que são de interesse do cliente. Nesta ótica, existem diversos modelos propostos para avaliação da qualidade de produtos de *software*. Este artigo baseia-se no modelo de qualidade ISO/IEC 9126-1.

4. O Modelo de Qualidade ISO/IEC 9126-1

A norma ISO/IEC 9126 apresenta conceitos e definições para qualidade de *software*, classificando a qualidade como externa e interna. São essas características que fazem com que os gerentes de projetos fiquem ou não satisfeitos com a qualidade dos produtos de *software* que são confeccionados por suas equipes de desenvolvimento. Marini (2002) afirma que o MEDE-PROS 01/97 é um método de avaliação da qualidade de *software* para fazer a avaliação da qualidade de *software* baseado na ISO/IEC 9126 e ISO/IEC 12119, sendo composto por: *Lista de Verificação, Manual do Avaliador e Modelo de Relatório de Avaliação*.

De acordo com Simão e Belchior (2003) existem seis características de qualidade para componentes de *software*: funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade, conforme mostra o Quadro 1.

Características de Qualidade de Software
Funcionalidade
Confiabilidade
Usabilidade
Eficiência
Manutenibilidade
Portabilidade

Quadro 1. Características da Qualidade para Componentes de *Software*

Fonte: Adaptado de Simão e Belchior, 2003.

Estas características, por sua vez, podem ser divididas da seguinte forma:

- ❶ Funcionalidade: Adequação, Acurácia, Autocontido, Coesão Funcional, Interoperabilidade, Segurança de Acesso e Conformidade com a Funcionalidade;
- ❷ Confiabilidade: Maturidade, Tolerância a Falhas, Recuperabilidade, Avaliabilidade e Conformidade com a Confiabilidade;

③ Usabilidade: Acessibilidade, Legibilidade, Inteligibilidade, Facilidade de Uso, Apreensibilidade, Operacionalidade, Atratividade e Conformidade com a Usabilidade;

④ Eficiência: Comportamento em Relação ao Tempo, Comportamento em Relação aos Recursos, Comportamento em Relação ao Estado, Escalabilidade, Nível de Granularidade e Conformidade com a Eficiência;

⑤ Manutenibilidade: Analisabilidade, Implementabilidade, Modificabilidade, Estabilidade, Testabilidade e Conformidade com a Manutenibilidade;

⑥ Portabilidade: Adaptabilidade, Capacidade de ser Instalado, Coexistência, Substituibilidade e Conformidade com a Portabilidade.

A Figura 1 ilustra o modelo de qualidade da ISO/IEC 9126-1 onde são descritas as seis características de qualidade para componentes de *software*.

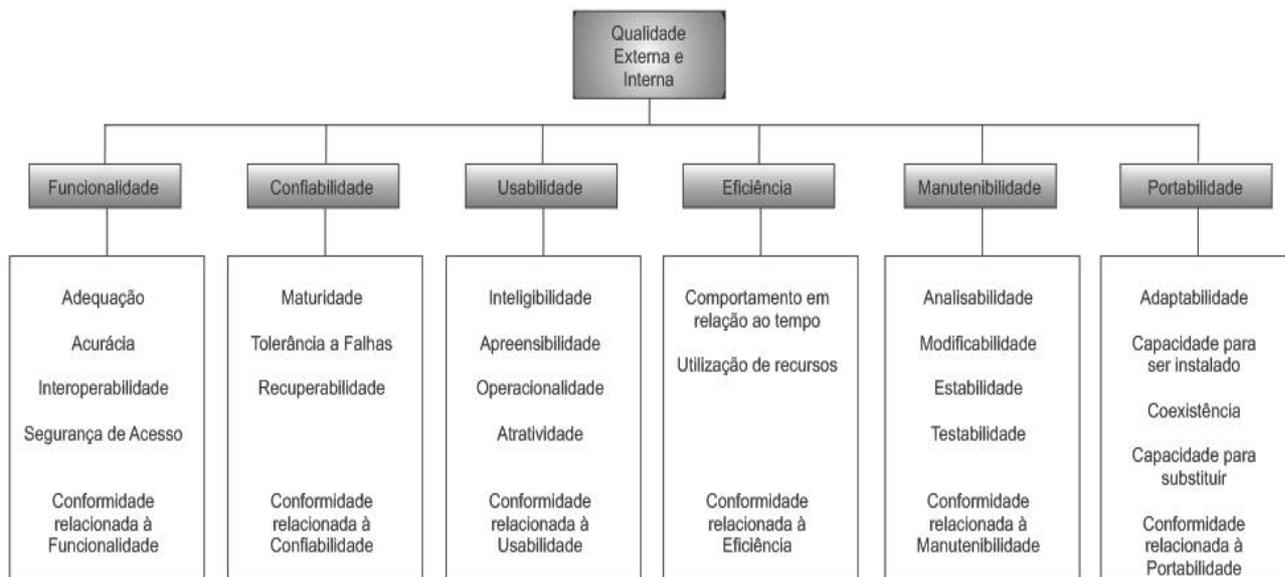


Figura 1. Modelo de Qualidade da ISO/IEC 9126-1.

Fonte: Adaptado de Koscianski e Soares, 2007.

Para avaliação dessa efetiva qualidade de *software*, esta pesquisa utilizou a teoria dos conjuntos *fuzzy* como ferramenta.

5. Teoria dos Conjuntos *Fuzzy*

Em 1965, o Professor Lotfi Zadeh formalizou o que, anos depois vinha a ser uma das maiores revoluções no setor matemático: a Lógica *Fuzzy* ou Lógica Nebulosa ou Lógica difusa. Esta teoria trata dos conjuntos não totalmente verdadeiros nem tampouco dos totalmente falsos. Em outras palavras, a lógica *fuzzy* deve ser vista como uma teoria matemática formal para a representação de incertezas (COSENZA et al., 2006, p. 2).

De acordo com Moré (2004, p. 47) a maior parte da linguagem natural contém ambigüidades e multiplicidade de sentidos. Em particular, os adjetivos que utilizamos para caracterizar objetos ou situações não nos permitem clareza suficiente, sendo ambíguos em termos de amplitude de significados.

A teoria dos conjuntos *fuzzy*, de acordo com Simões e Shaw (2007, p. 21) é baseada no fato de que os conjuntos existentes no mundo real não possuem limites precisos. Os conjuntos *fuzzy* podem ser vistos como uma generalização da noção de conjunto na qual a função de pertinência pode assumir valores no intervalo [0,1] (FARIA, et al., 2008, p. 5). Segundo Goldschmidt e Passos (2005) explicam que existem três formas de definir conjuntos na teoria de conjuntos *fuzzy*: representação explícita, representação implícita e representação pela função características.

A Figura 2 ilustra o modelo *fuzzy* utilizado nesta pesquisa.

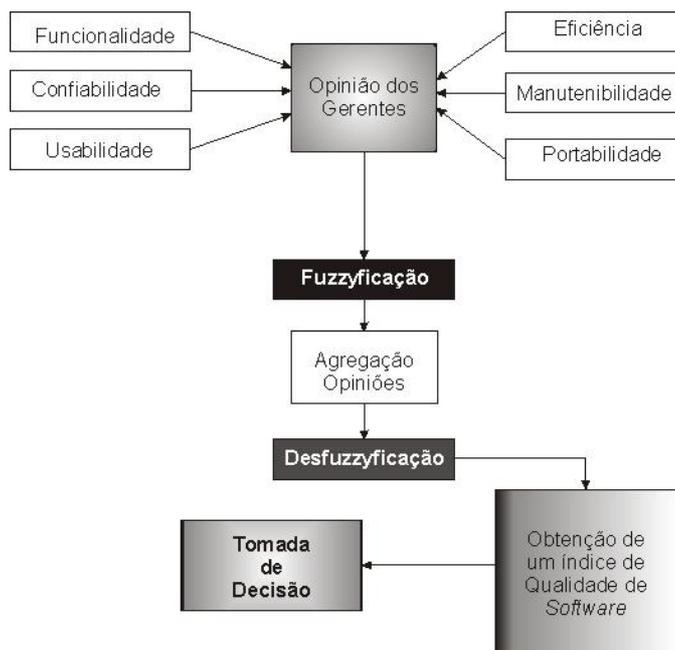


Figura 2. Descrição do Modelo Fuzzy Proposto.

Fonte: Elaboração própria.

Os conjuntos *fuzzy* prestam-se às representações de conceitos vagos, expressados na linguagem natural, dependendo do contexto em que são usados, como por exemplo, aqueles aplicados no modelo utilizado (BELCHIOR, 1997).

6. Metodologia

Para a aplicação do modelo proposto e ilustrado na Figura 2, foi necessário utilizar oito etapas conforme descrição a seguir.

Primeira etapa: Determinação das variáveis lingüísticas do modelo. Nesta etapa as variáveis lingüísticas foram determinadas através de uma revisão bibliográfica que permitiu identificar seis construtos de qualidade de *software* - funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade.

Segunda etapa: Escolha dos termos lingüísticos a serem utilizados para as medições a serem realizadas. Para medir o quão presente estão os critérios de qualidade nos produtos de *software* desenvolvidos numa fundação pública estadual foram escolhidos cinco termos lingüísticos: *totalmente ausente*, *baixa presença*, *moderadamente presente*, *altamente presente* e *totalmente presente* (ver o Quadro 2).

Escala	Equivalência	Descrição
0	Totalmente ausente	Indica total ausência do critério de qualidade avaliado
1	Baixa presença	Indica um baixo grau de presença do critério de qualidade avaliado
2	Moderadamente presente	Indica um grau de presença moderada do critério.
3	Altamente presente	Indica um alto grau de presença do critério, mas não de forma plena.
4	Total Presença	Indica que não há dúvidas de que o critério está totalmente presente.

Quadro 2. Escala utilizada para medição dos graus de presença dos critérios de qualidade nos produtos de *software*

Fonte: Elaboração própria

Para medir o quão importante resulta a presença dos critérios de qualidade nos produtos de *software* desenvolvidos, foram escolhidos também cinco termos lingüísticos: *sem importância, pouco importante, moderadamente importante, importante e muito importante* (ver o Quadro 3).

Escala	Equivalência	Descrição
0	Sem importância	Indica que o critério que está sendo apresentado não tem nenhuma importância.
1	Pouco importante	Indica que o critério que está sendo apresentado tem pouca importância.
2	Moderadamente importante	Indica que o critério que está sendo apresentado tem importância em algumas circunstâncias mas nem sempre.
3	Importante	Indica que o critério que está sendo apresentado é importante.
4	Muito importante	Indica que não há dúvidas com que o critério que está sendo apresentado é imprescindível.

Quadro 3. Escala utilizada para medição dos graus de importância de critérios de qualidade de produtos de *software*

Fonte: Elaboração própria

Os quadros 2 e 3 apresentam os valores atribuídos às escalas utilizadas para os questionários estruturados utilizados.

Terceira etapa: Elaboração dos questionários estruturados. Nesta etapa foram elaborados três questionários com o objetivo de levantar o grau de presença e importância de critérios de qualidade de *software* e determinar o peso de cada um dos gerentes de projetos de *software*.

Quarta etapa: Criação das funções de pertinências para os termos *fuzzy* apresentados na segunda etapa. Para representar as avaliações imprecisas e subjetivas nas opiniões dos gerentes foram escolhidos os conjuntos *fuzzy* triangulares pela capacidade que possuem de representar essa incerteza e para que seja computada

facilmente pelo computador. Nas figuras 3 e 4 aparecem representadas as funções de pertinências. Os conjuntos *fuzzy* correspondentes aos graus de presença de critérios de qualidade de *software* foram: *totalmente ausente (TA)*, *baixa presença (BP)*, *moderadamente presente (MoP)*, *altamente presente (AP)* e *total presença (TP)*.

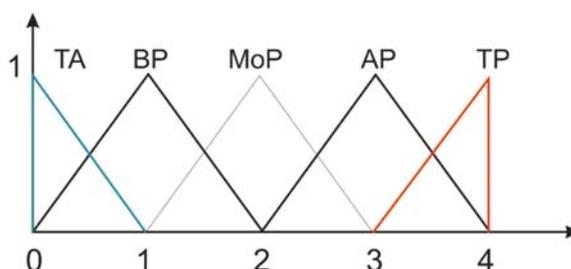


Figura 3. Conjuntos *Fuzzy* dos termos relacionados aos graus de presença de critérios de qualidade de *software*

Fonte: Elaboração própria

Os conjuntos *fuzzy* correspondentes aos graus de importância dos critérios de qualidade nos produtos de *software* foram *muito importante (MI)*, *importante (I)*, *moderadamente importante (MoI)*, *pouco importante (PI)* e *sem importância (SI)*.

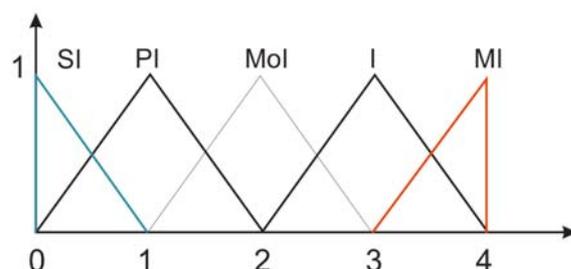


Figura 4. Conjuntos *Fuzzy* dos termos relacionados aos graus de importância dos critérios de qualidade de *software*

Fonte: Elaboração própria

O Quadro 4 ilustra os números triangulares *fuzzy* correspondentes aos conjuntos *fuzzy* escolhidos.

Valor do termo <i>fuzzy</i>	Nº triangular <i>fuzzy</i>	Grau de Presença	Grau de Importância
4	(3, 4, 4)	Total Presença (TP)	Muito Importante (MI)
3	(2, 3, 4)	Alta Presença (AP)	Importante (I)
2	(1, 2, 3)	Moderada Presença (MoP)	Moderadamente Importante (MoI)
1	(0, 1, 2)	Baixa Presença (BP)	Pouco Importante (PI)
0	(0, 0, 1)	Totalmente Ausente (TA)	Sem Importância (SI)

Quadro 4. Números *fuzzy* triangulares correspondentes aos conjuntos *fuzzy* escolhidos

Fonte: Elaboração própria

Quinta etapa: Aplicação dos questionários estruturados aos gerentes de projetos de *software*. Em seguida os questionários foram aplicados a 12 gerentes de projetos de *software* lotados na divisão de informática da fundação pública estadual estudada.

Sexta etapa: Para identificar os pesos dos gerentes de projetos de *software* (coeficiente de importância) foram usadas diferentes escalas de medição, atribuídas de acordo com os itens de interesse levantados (nível de experiência, prática, nível de conhecimento, grau de instrução, participação em projetos, participação em congressos e currículo do gerente). Cada item avaliado recebeu uma pontuação subjetiva entre 0 e 1. Na Tabela 1 aparecem os valores dados a cada gerente (G01, G02, G03,..., G12). O coeficiente de importância (peso dos gerentes) é calculado a partir da divisão da somatória das pontuações subjetivas recebidas por item pela somatória total das pontuações recebidas de todos os gerentes ($\Sigma=64,00$), segundo a fórmula a seguir:

$$CI_{gerente_i} = \frac{\sum_{i=1}^7 \text{Avaliações Itens Gerente}_i}{\sum_{i=1}^{12} \text{Avaliações Gerentes}}$$

Tabela 1. Levantamento do coeficiente de importância de cada gerente de projeto

Peso Empregado aos Gerentes de Projetos												
Itens	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12
Experiência	0,75	1,00	1,00	0,75	1,00	1,00	0,50	0,75	0,50	0,50	0,75	0,50
Prática	1,00	1,00	1,00	1,00	1,00	1,00	0,75	1,00	0,75	1,00	0,75	0,50
Conhecimento	1,00	1,00	1,00	0,75	0,75	1,00	0,75	1,00	0,50	1,00	0,75	0,50
Grau Instrução	1,00	0,75	0,50	0,75	1,00	0,50	0,50	0,25	0,25	0,50	0,75	0,25
Participação em Projetos	1,00	0,75	1,00	0,75	1,00	1,00	0,75	0,50	0,50	1,00	1,00	0,75
Participação em Congressos	0,75	1,00	0,50	0,50	0,75	0,25	0,50	0,25	0,25	0,75	0,50	0,50
Currículo	1,00	1,00	0,75	0,75	1,00	1,00	0,75	1,00	0,75	1,00	1,00	0,75
$\Sigma= 64,00$	6,50	6,50	5,75	5,25	6,50	5,75	4,50	4,75	3,50	5,75	5,50	3,75
Peso dos gerentes ($\Sigma=1,00$)	0,10	0,10	0,09	0,08	0,10	0,09	0,07	0,07	0,05	0,09	0,09	0,06

Fonte: Elaboração própria

O Gráfico 1, elaborado a partir dos coeficientes de importância (pesos) obtidos ilustra os pesos atribuídos aos gerentes de projetos de *software*.

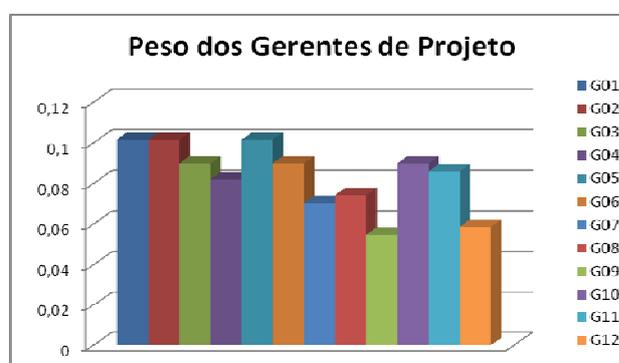


Gráfico 1. Pontuações *fuzzy* (entre 0 e 1) associadas aos gerentes de projetos

Fonte: Elaboração própria.

Sétima etapa: Coleta dos dados. As respostas obtidas (*fuzzificação*) a partir dos questionários aplicados foram coletadas e tabuladas.

Oitava etapa: Tratamento dos dados. Na fase de agregação das opiniões *fuzzy* foi utilizada a fórmula da média *fuzzy*. Neste caso considerou-se a avaliação *fuzzy* de cada gerente por critério $(Aval)_{crit}$, ponderada pelo coeficiente de importância de cada gerente (CI_{ger}) , obtendo-se assim os triângulos *fuzzy* agregados $(a,m,b)_{agreg}$.

$$(a, m, b)_{agreg_j} = \sum_{i=1}^{12} \sum_{j=1}^{18} CI_{ger_i} * (Aval)_{crit_j}$$

Os três valores que formam cada triângulo *fuzzy* têm diferentes significados: o valor do extremo esquerdo representa o menor valor com a menor possibilidade de pertencer ao conjunto *fuzzy*; o valor central é o valor com maior possibilidade de pertencer ao conjunto *fuzzy* e o valor da extrema direita representa o maior valor com a menor possibilidade de pertencer ao conjunto *fuzzy*.

Para se obter um valor preciso ou *crisp* representativo por cada conjunto *fuzzy* (*defuzzificação*), ou seja, um único valor numérico discreto que melhor representasse o valor *crisp* inferido das variáveis lingüísticas utilizou-se a seguinte fórmula:

$$V_{crisp} = \frac{(a + 2m + b)_{agreg_j}}{4}$$

Onde a, m e b são os valores do triângulo *fuzzy* obtido a partir da agregação das opiniões dos gerentes. Cada valor *crisp* obtido foi dividido pelo valor máximo entre todos os valores *crisp* (processo de normalização), isto é, o valor normalizado calculado de acordo com a fórmula a seguir:

$$V_{norm} = \frac{V_{crisp}}{V_{max}}$$

Uma vez conhecidos os valores *crisp* foram calculadas as distâncias (*gap*) entre as demandas de qualidade dos produtos de *software* (graus de importância dos critérios de qualidade) e o atendimento a essas exigências pela instituição (graus de presença desses critérios de qualidade). Para isto, subtraímos do valor *crisp* correspondente ao grau de presença o valor *crisp* referente ao grau de importância, para cada um dos dezoito critérios apresentados (ver fórmula a seguir).

$$D_{crisp} = V_{crisp} P - V_{crisp} I$$

Na Tabela 2 aparecem representados os valores de distância (*gap*). Uma distância negativa significa que a oferta de qualidade dos produtos de *software* desenvolvidos não cobrem as necessidades ou expectativas.

Tabela 2. Distância entre as demandas de critérios de qualidade dos produtos de *software*

Critérios	Pres.	Imp.	Gap	Critérios	Pres.	Imp.	Gap
Q01	2,74	2,83	-0,09	Q22	2,15	2,84	-0,69
Q02	2,41	2,84	-0,43	Q23	2,42	3,10	-0,68
Q03	2,33	2,78	-0,45	Q24	2,81	3,14	-0,33

Q04	2,58	2,99	-0,41	Q25	2,69	3,06	-0,37
Q05	2,81	2,99	-0,18	Q26	2,77	2,97	-0,2
Q06	2,67	3,13	-0,46	Q27	2,31	3,06	-0,75
Q07	2,77	3,14	-0,37	Q28	2,59	3,00	-0,41
Q08	2,80	3,13	-0,33	Q29	2,38	2,40	-0,02
Q09	2,70	3,20	-0,5	Q30	2,76	2,77	-0,01
Q10	2,68	2,89	-0,21	Q31	2,92	2,61	0,31
Q11	2,42	3,04	-0,62	Q32	2,58	1,18	1,4
Q12	1,92	2,67	-0,75	Q33	1,64	1,49	0,15
Q13	2,59	3,04	-0,45	Q34	3,03	3,07	-0,04
Q14	2,44	3,03	-0,59	Q35	2,47	3,13	-0,66
Q15	2,12	3,25	-1,13	Q36	2,49	3,19	-0,7
Q16	2,45	2,96	-0,51	Q37	2,29	3,19	-0,9
Q17	2,51	3,13	-0,62	Q38	2,04	3,12	-1,08
Q18	1,78	2,84	-1,06	Q39	2,14	3,03	-0,89
Q19	1,66	2,84	-1,18	Q40	2,62	3,03	-0,41
Q20	2,65	2,99	-0,34	Q41	2,68	2,84	-0,16
Q21	2,48	3,03	-0,55				

Fonte: Elaboração própria.

Isto resulta num hiato e nesse caso é preciso aplicar recursos organizacionais com o objetivo de melhorar o critério de qualidade. Já uma distância positiva significa que a oferta supera a demanda e nesse caso não é preciso aplicar recursos. Graficamente é possível visualizar estas distâncias através do Gráfico 2.

Representação das Distâncias

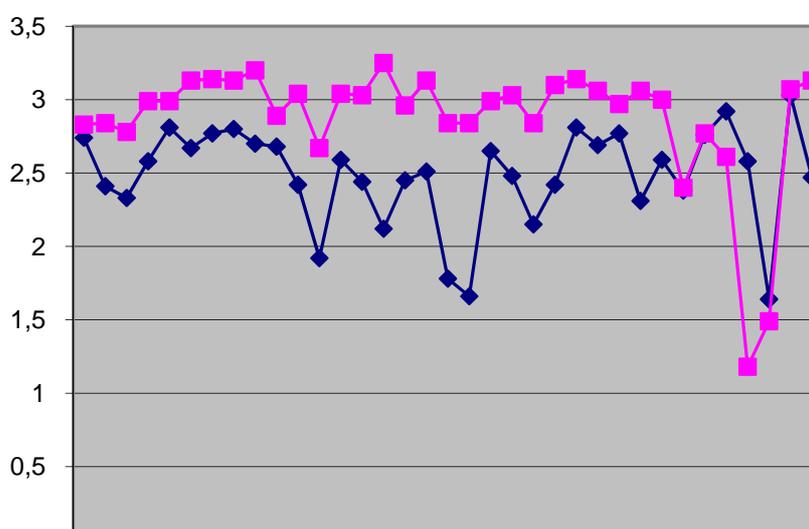


Gráfico 2. Grau de presença *versus* grau de importância dos critérios de qualidade de *software*

Fonte: Elaboração própria.

Muitas vezes o valor de distância (*gap*) não constitui informação relevante no momento de priorizar os recursos para investir em melhorias de critérios. Nesse caso o grau de semelhança entre conjuntos *fuzzy* (importância *versus* presença) vem a complementar o processo de tomada de decisão. Sendo assim buscou-se identificar o grau de semelhança entre os conjuntos *fuzzy* de importância e presença para cada um dos critérios de qualidade de produtos de *software* utilizando para isto a seguinte fórmula:

$$G_{sem}(\tilde{A}, \tilde{B}) = \frac{AI}{AT} = \frac{\min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))}{\max(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))}$$

Onde \tilde{A} é o triângulo *fuzzy* agregado do conjunto critérios de qualidade, \tilde{B} é o triângulo *fuzzy* agregado do conjunto importância de critérios de qualidade, AI é a área de interseção dos triângulos *fuzzy* (\tilde{A}, \tilde{B}) e AT é a área total dos triângulos *fuzzy* (\tilde{A}, \tilde{B}). A Tabela 3 ilustra os resultados obtidos.

Tabela 3. Semelhança existente entre os graus de presença e os graus de importância dos critérios de qualidade de produtos de *software*

Item	Nº Triangular Fuzzy			Nº Triangular Fuzzy			Área de	Área	Grau
	I - Presença			II - Importância			Interseção	Total	Semelhança
Q01	1,77	2,77	3,66	1,88	2,88	3,68	0,838201	1,006799	0,833
Q02	1,43	2,43	3,33	1,86	2,86	3,77	0,568658	1,336342	0,426
Q03	1,33	2,33	3,33	1,80	2,80	3,71	0,585225	1,369775	0,427
Q04	1,58	2,58	3,58	2,07	3,07	3,77	0,570025	1,279975	0,445
Q05	1,90	2,90	3,52	2,09	3,09	3,70	0,631142	0,983858	0,641
Q06	1,70	2,70	3,59	2,24	3,24	3,77	0,482143	1,227857	0,393
Q07	1,83	2,83	3,58	2,29	3,29	3,71	0,475457	1,109543	0,429
Q08	1,84	2,84	3,67	2,25	3,25	3,77	0,550929	1,124071	0,490
Q09	1,77	2,77	3,49	2,34	3,34	3,77	0,384448	1,190552	0,323
Q10	1,68	2,68	3,68	1,96	2,96	3,67	0,739600	1,115400	0,663
Q11	1,52	2,45	3,26	2,11	3,11	3,84	0,365331	1,369669	0,267
Q12	1,08	1,91	2,77	1,67	2,67	3,67	0,325269	1,519731	0,214
Q13	1,68	2,68	3,30	2,13	3,13	3,77	0,422500	1,207500	0,350
Q14	1,52	2,44	3,35	2,12	3,12	3,77	0,396047	1,343953	0,295
Q15	1,28	2,11	2,97	2,40	3,40	3,77	0,087339	1,442661	0,061
Q16	1,52	2,45	3,39	2,05	3,05	3,67	0,462784	1,282216	0,361
Q17	1,60	2,53	3,37	2,24	3,24	3,77	0,346984	1,303016	0,266
Q18	0,84	1,78	2,70	1,86	2,86	3,77	0,183750	1,701250	0,108
Q19	0,81	1,64	2,53	1,86	2,86	3,77	0,118757	1,696243	0,070
Q20	1,69	2,69	3,54	2,07	3,07	3,77	0,584027	1,190973	0,490

Q21	1,53	2,53	3,34	2,11	3,11	3,77	0,417928	1,317072	0,317
Q22	1,15	2,15	3,15	1,93	2,93	3,59	0,372100	1,457900	0,255
Q23	1,46	2,46	3,30	2,21	3,21	3,77	0,322853	1,377147	0,234
Q24	1,90	2,90	3,52	2,26	3,26	3,77	0,490000	1,075000	0,456
Q25	1,75	2,75	3,50	2,16	3,16	3,77	0,513029	1,166971	0,440
Q26	1,86	2,86	3,50	2,07	3,07	3,69	0,623445	1,006555	0,619
Q27	1,31	2,31	3,31	2,16	3,16	3,77	0,330625	1,474375	0,224
Q28	1,68	2,59	3,48	2,10	3,10	3,71	0,503810	1,201190	0,419
Q29	1,50	2,42	3,17	1,40	2,40	3,40	0,645514	1,189486	0,543
Q30	1,79	2,79	3,67	1,84	2,84	3,55	0,890665	0,904335	0,985
Q31	1,96	2,96	3,77	1,64	2,64	3,54	0,656947	1,198053	0,548
Q32	1,74	2,64	3,29	0,59	1,04	2,04	0,023684	1,476316	0,016
Q33	0,79	1,59	2,59	0,86	1,40	2,31	0,750577	0,874423	0,858
Q34	2,09	3,09	3,84	2,17	3,17	3,77	0,796829	0,878171	0,907
Q35	1,58	2,54	3,21	2,25	3,25	3,77	0,275928	1,299072	0,212
Q36	1,49	2,49	3,49	2,33	3,33	3,77	0,336400	1,383600	0,243
Q37	1,39	2,28	3,22	2,33	3,33	3,77	0,204149	1,430851	0,143
Q38	1,19	2,03	2,93	2,23	3,23	3,77	0,128947	1,511053	0,085
Q39	1,21	2,12	3,12	2,09	3,09	3,84	0,265225	1,564775	0,169
Q40	1,64	2,64	3,55	2,09	3,09	3,84	0,558010	1,271990	0,439
Q41	1,68	2,68	3,68	1,88	2,88	3,70	0,810000	1,100000	0,736

Fonte: Elaboração própria.

O Gráfico 3, ilustra a variação do grau de semelhança entre a demanda e a oferta por critério de qualidade de produtos de *software*.

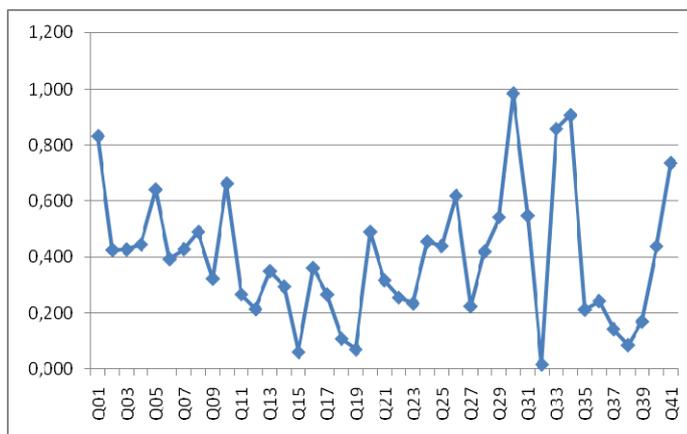


Gráfico 3. Representação do grau de semelhança entre o conjunto *fuzzy* presença e o conjunto *fuzzy* importância para cada critério de qualidade de produtos de *software*

Fonte: Elaboração própria.

7. Análise de Resultados e Tomada de Decisão

Análise dos resultados e tomada de decisão. A partir dos valores normalizados, um índice de qualidade de produtos de *software* foi obtido utilizando a média ponderada dada pela fórmula a seguir:

$$I_{Qual} = \frac{\sum_{i=1}^{41} (G_{IMP_i} * G_{PRES_i})}{\sum_{i=1}^{41} G_{IMP_i}} = 0,82$$

Existe uma possibilidade alta (0,82) de que os critérios de qualidade estejam presentes nos produtos de *software* que são desenvolvidos tomando em conta os graus de importância (G_{imp}) e presença (G_{pres}) desses critérios. Pelos resultados observa-se que as maiores distâncias aparecem nos critérios Q19 - Localizabilidade Externa (0,35), Q32 - Existência de Capital (0,49) e Q38 - Adequabilidade de Cronograma (0,29). São estas as variáveis que merecem uma atenção especial, por parte do diretor da diretoria de informática nessa fundação pública estadual, na tomada de decisão para minimizar as possibilidades de produzir produtos de *software* com baixa ou nenhuma qualidade efetivamente.

Ao calcular o grau de semelhança existente entre os conjuntos *fuzzy* “presença” e os conjuntos *fuzzy* “importância” dos critérios de qualidade de produtos de *software*, obteve-se para o item Q30 – Relevância de Benefícios (característica que avalia se as estimativas de benefícios tangíveis e intangíveis são aceitas como relevantes por usuários e desenvolvedores) o maior grau de semelhança (0,985) e para o item Q32 – Existência de Capital (característica que avalia se a organização possui capital suficiente para custear o desenvolvimento de produtos de software) o menor grau de semelhança (0,016).

Constata-se que independentemente do índice ser elevado (0,82) ainda falta um caminho por percorrer (0,18) para chegar ao “ótimo”. Neste sentido haverá necessidade de aplicar recursos nos itens Q15 (acessibilidade – característica que avalia se qualquer usuário autorizado pode facilmente consultar a especificação e/ou obter uma cópia da mesma), Q18 (localizabilidade interna – característica que avalia se existem facilidades para se localizar todos os elementos, dentro de uma especificação, relacionados com determinado aspecto ou assunto), Q19 (localizabilidade externa - característica que avalia se existem facilidades para se localizar todas as especificações e demais documentos relacionados a um determinado aspecto ou assunto), Q35 (disponibilidade de tecnologia – característica que avalia se a equipe encarregada do desenvolvimento tem disponível a tecnologia necessária para conduzir o desenvolvimento), Q37 (disponibilidade de mão de obra – característica que avalia se estão disponíveis os recursos humanos com o conhecimento e experiência necessária para a realização do desenvolvimento e operação do software), Q38 (adequabilidade do cronograma – característica que avalia se o software pode ser construído no tempo previsto pelo cronograma, considerando as possíveis ocorrências de imprevistos e sem descuidar da qualidade definida para o produto) e Q39 (flexibilidade de cronograma – característica que avalia se o cronograma aceito para o desenvolvimento pode atender, na medida do possível fatores tais como introdução de atividades não projetadas, contingências etc.), apresentam insuficiência de recursos ou falta de informação, o que requer um

tratamento minucioso em busca de melhorar o grau de presença destes critérios de qualidade nos produtos de *software* desenvolvidos pela fundação pública.

8. Considerações Finais

Este artigo tratou a teoria dos conjuntos *fuzzy* como ferramenta para avaliação da qualidade de produtos de *software* numa fundação pública estadual. Isso foi possível a partir de um conjunto de 41 critérios de qualidade de produtos de *software*.

A abordagem *fuzzy* permitiu tratar de forma matematicamente sólida, medidas subjetivas sujeitas a incertezas, obtidas a partir da opinião pessoal dos 12 gerentes de projetos respondentes.

Foi obtido um índice de qualidade de produtos de *software* (0,82) considerado muito bom. Apesar disso, foram identificadas lacunas não tão satisfatórias, o que permitiu aprimorar o processo de tomada de decisão dessa fundação pública estadual.

Referências

- BELCHIOR, A.D. **Um Modelo Fuzzy para Avaliação da Qualidade de Software**. Tese de Doutorado, Engenharia de Sistemas e Computação, COPPE, Universidade Federal do Rio de Janeiro, UFRJ, RJ, 1997.
- BOENTE, A.N.P. **Gerenciamento e Controle de Projetos**. Rio de Janeiro: Axcel Books, 2003.
- BOENTE, A.N.P.; OLIVEIRA, F.S.G.; ALVES, J.C.N. **RUP como Metodologia de Desenvolvimento de Software para Obtenção da Qualidade de Software**. Anais: V Simpósio de Excelência em Gestão e Tecnologia. out. 2008.
- COSENZA, H.J.S.R. et al. **Aplicação de Um Modelo de Hierarquização como Instrumento para Tomada de Decisão**: Caso de uma Multinacional. In: XXVI Encontro Nacional de Engenharia de Produção, ENEGEP, 2006, Fortaleza, 2006.
- FARIA, M.N. et al. **Um Sistema de Avaliação em EAD baseado em Lógica Fuzzy**. Revista Eletrônica Horizonte Científico. Minas Gerais, edição 2008, dez/2008.
- GOLDSCHMIDT, R.R.; PASSOS, E. **Data Mining**: Um Guia Prático. Rio de Janeiro: Campus, 2005.
- JURAN, J.M. **Controle da Qualidade**. São Paulo: Makron Books, 1991.
- KOSCIANSKI, A.; SOARES, M.S. **Qualidade de Software** - Aprenda as Metodologias e Técnicas mais Modernas para o Desenvolvimento de Software. 2. ed. São Paulo: Novatec, 2007.
- MARINI, M.J. **Uma Ferramenta de Suporte à Avaliação da Qualidade de Software de Aplicativos Voltados à Gestão Empresarial**. Dissertação de Mestrado, Ciência da Computação, Sistemas de Computação, Universidade Federal de Santa Catarina, SC, 2002
- MECENAS, I.; OLIVEIRA, V. **Qualidade de Software** – Uma Metodologia para Homologação de Sistemas. Rio de Janeiro: Alta Books, 2005.

MORÉ, J.D. **Aplicação da lógica *Fuzzy* na avaliação da confiabilidade humana nos ensaios não destrutivos por ultra-som**. Tese de Doutorado submetida ao programa de pós-graduação de Engenharia Metalúrgica e dos Materiais da Universidade Federal do Rio de Janeiro, COPPE, 2004.

POPADIUK, S.; SANTOS, C.R. dos. **Fatores de Influência na Adoção da Metodologia de Gestão de Projetos em TI**: Uma Comparação entre Usuários e Potenciais Usuários Mediante o uso do SmartPLS. Revista Eletrônica de Administração. Rio Grande do Sul, edição 59, volume 14, n° 1, jan-abr/2008.

PRESSMAN, R.S. **Engenharia de *Software***. 6. ed. São Paulo: McGraw-Hill, 2006.

REZENDE, D.A. **Engenharia de *Software* e Sistemas de Informação**. Rio de Janeiro: Brasport, 1999.

ROCHA, A.R.C. da.; *et al.* **Uma Experiência na Definição do Processo de Desenvolvimento e Avaliação de *Software* segundo a Norma ISO**, Relatório Técnico ES-302/94, COPPE/UFRJ, jun. 1994.

SIMÃO, R.P.S.; BELCHIOR, A.D. **Componentes de Negócio**: Uma Avaliação das Características de Qualidade. In: Conferência IADIS Íbero Americana WWW/Internet, 2003.

SIMÕES, M.G. e SHAW, I.S. **Controle e Modelagem *Fuzzy***. 2. ed. Revisada e Completa. São Paulo: Blucher: FAPESP, 2007.

SOARES, M.S. **Metodologias Ágeis Extreme Programming e Scrum para o Desenvolvimento de *Software***. Revista Eletrônica de Sistemas de Informação. vol. 3, p. 8-13, 2004.

TSUKUMO, A. N. *et al.* **Qualidade de *Software***: Visões de Produto e Processo de *Software*. Publicado na II Escola Regional de Informática da Sociedade Brasileira de Computação Regional de São Paulo. p. 173-189. Piracicaba, SP, 1997.

WEBER, K. C.; ROCHA, A. R. C. da. **Qualidade e Produtividade em *Software***. 3. ed. São Paulo: Makron Books, 1999.