

## UM MODELO FUZZY PARA AVALIAÇÃO DA QUALIDADE DE SOFTWARE

Arnaldo Dias Belchior<sup>1,3,4</sup> Geraldo Bonorino Xexéo<sup>1,2</sup> Ana Regina Cavalcanti da Rocha<sup>1</sup>

<sup>1</sup> Universidade Federal do Rio de Janeiro - COPPE/Sistemas

<sup>2</sup> Instituto de Matemática / UFRJ

Caixa Postal 68511

21945-970 Rio de Janeiro - RJ

<sup>3</sup> Banco do Nordeste do Brasil

<sup>4</sup> Universidade Estadual do Ceará (UECE)

Fortaleza - CE

belchior@bnb.gov.br xexeo@cos.ufrj.br darocha@cos.ufrj.br

### Resumo

Este trabalho desenvolve um modelo *fuzzy* para a avaliação da qualidade de software, a partir da extensão de um modelo para avaliação da qualidade, anteriormente proposto, e utiliza, para isto, conceitos e propriedades da teoria dos conjuntos *fuzzy*. O modelo proposto possui cinco etapas, para a execução de seus objetivos, podendo envolver três situações distintas. A primeira, objetiva obter um padrão de qualidade para o produto de software, ou para o domínio de aplicação considerado. A segunda, avalia a qualidade de um produto de software, apoiando-se em um padrão de qualidade já definido, anteriormente, para esse produto ou seu domínio de aplicação. A terceira, estima a qualidade de um produto de software, quando não há um padrão de qualidade disponível. Uma experiência com a avaliação da qualidade de especificações de software foi realizada através deste modelo *fuzzy* com o objetivo de validá-lo e exemplificar seu uso. Os resultados experimentais obtidos corroboraram com os resultados previstos axiomáticamente.

**Palavras-chave:** avaliação da qualidade de software, métricas, teoria *fuzzy*.

### Abstract

This work presents a fuzzy model to software quality evaluate, extending a software quality model previously proposed, and using to do this concepts and properties of fuzzy sets theory. The proposed model has five steps to do its objectives, embracing three different situations. The first aims to get a quality standard to a software product or application domain considered. The second evaluates the quality of a software product, supported by a quality standard previously defined to this product or its application domain. The third assesses the quality of a software product, when there is no quality standard available. An experience with evaluation of software quality specification was realized through this fuzzy model intending to validate them and to exemplify its use. The experimental results agreed with those predicted axiomatically.

**Keywords:** software quality evaluation, metrics, fuzzy theory.

## I - INTRODUÇÃO

Uma estrutura de avaliação de software tem por objetivo estimar a sua qualidade, através de um conjunto básico de atributos, evidenciando seus aspectos relevantes. Para isto, as informações sobre o objeto da avaliação devem estar dispostas de forma organizada, onde características específicas do software possam ser identificadas, para otimizar o processo de tomada de decisão (BOLOIX, 1995). A tomada de decisão pode ser vista como um processo de seleção de alternativas "suficientemente boas" ou da escolha de cursos de ações, para a obtenção de um objetivo. Esse processo envolve incertezas e, portanto, é necessário que se tenha a habilidade de manusear informações imprecisas e vagas, levando-se em consideração diferentes visões, atitudes e crenças das partes envolvidas (RIBEIRO, 1996). Portanto, é importante fazer a conexão entre a informação e a decisão de um indivíduo, que deve escolher uma ação entre muitas possíveis, em áreas distintas (SIMONELLI, 1996).

Uma vez que o processo de tomada de decisão é centrado em pessoas humanas, como também o é o processo de avaliação de software, com suas inerentes subjetividades e inconsistências na definição do problema, os conjuntos *fuzzy* são potencialmente adequados nesta área, pois (IBRAHIM, 1992): (i) possuem a habilidade de representar atributos; (ii) detêm formas convenientes e avaliáveis para a combinação de atributos, que podem estar vaga ou precisamente definidos, e (iii) manuseiam diferentes graus de importância para cada atributo considerado.

A princípio, pode parecer incoerente a utilização da teoria *fuzzy* (nebulosa) para a avaliação da qualidade de software. Para justificar esta escolha, cita-se a argumentação de ZIMMERMAN (1991): "*A teoria dos conjuntos fuzzy fornece uma estrutura matemática estrita (não há nada de nebuloso na teoria dos conjuntos fuzzy!), na qual um fenômeno conceitual vago pode ser precisa e rigorosamente estudado*".

Como em muitas outras áreas do conhecimento humano, a avaliação da qualidade de software envolve a apreciação de múltiplos atributos, através do julgamento de um grupo de especialistas. Cada especialista tem a sua própria opinião e estima um grau de importância para cada atributo julgado, segundo sua percepção ou o seu nível de entendimento da questão proposta. Daí o interesse de se conseguir um processo de agregação, que consolide o consenso dos especialistas envolvidos na avaliação.

No processo de avaliação da qualidade de software, não basta apenas identificar que atributos determinam essa qualidade, mas também que procedimentos adotar, para controlar seu processo de desenvolvimento de forma a atingir o nível de qualidade desejado. Isto é realizado através da aplicação de métricas de forma organizada e projetada, tornando os desenvolvedores mais conscientizados da relevância do gerenciamento e dos compromissos para com a qualidade.

O uso de métricas de qualidade de software (FENTON, 1997, KITCHENHAM, 1996, PRATHER, 1996, MÖLLER, 1993, SCHNEIDEWIND, 1992) deve estar apoiado em um método para controle da qualidade, para que se alcance, convenientemente, os objetivos almejados. Neste trabalho, o método escolhido para a avaliação da qualidade de software é o *Modelo Rocha* (ROCHA, 1983), por já ter sido utilizado satisfatória e eficazmente em uma grande quantidade de domínios de aplicação. Para alinhar a confiabilidade desse modelo às propriedades da teoria *fuzzy*, propõe-se uma extensão do mesmo, para suportar o *Modelo Fuzzy para Avaliação da Qualidade de Software* (BELCHIOR, 1997).

## II - EXTENSÃO DO MODELO ROCHA

O Modelo Rocha (ROCHA, 1983) define qualidade de software a partir dos seguintes conceitos:

- **Objetivos da qualidade:** são as propriedades gerais, que o produto deve possuir.
- **Fatores de qualidade:** determinam a qualidade na visão dos diferentes usuários do produto, e podem ser compostos por *subfatores*, quando estes não definem completamente, por si só, um objetivo.
- **Crítérios:** são atributos primitivos, possíveis de serem avaliados.
- **Processos de avaliação:** determinam as métricas e os instrumentos a serem utilizados na avaliação, para se medir o grau de presença, no produto, de um determinado critério.
- **Medidas:** são o resultado da avaliação do produto, segundo os critérios.
- **Medidas agregadas:** são o resultado da agregação das medidas obtidas ao se avaliar de acordo com os critérios, e quantificam os fatores.

Os objetivos de qualidade são atingidos através dos fatores de qualidade, que podem ser compostos por subfatores. Objetivos, fatores e subfatores não são diretamente mensuráveis e só podem ser avaliados através de critérios. Nenhum critério isolado é uma descrição completa de um determinado fator ou subfator. Da mesma maneira, nenhum fator define completamente um objetivo.

O Modelo Rocha não fornecia formas adequadas para se fazer a agregação de medidas. Definiu-se, então, uma extensão desse modelo através da utilização de conceitos e propriedades da teoria dos conjuntos *fuzzy* e, assim, o *Modelo Rocha Estendido* foi dotado da potencialidade dessa teoria em mapear modelos qualitativos de tomada de decisão, e da consistência no tratamento de incertezas e na agregação de informações. Neste contexto, alguns conceitos do Modelo Rocha foram estendidos (➤) e outros foram acrescidos (⊛) (figura 1) (BELCHIOR, 1997)

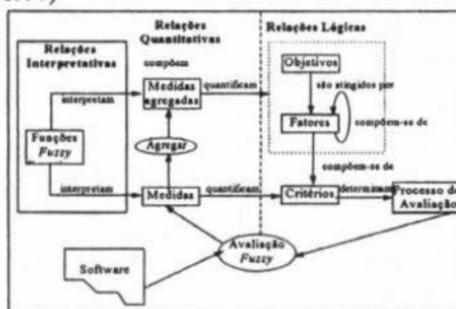


Figura 1: Modelo Rocha Estendido

➤ **Medidas:** são o resultado da avaliação do produto, segundo os critérios, através de um conjunto termos lingüísticos *fuzzy*, mapeados por números *fuzzy*.

➤ **Medidas agregadas:** são o resultado da agregação das medidas obtidas ao se avaliar de acordo com os critérios. São, também, o resultado da agregação de critérios em subfatores, fatores, objetivos, e no valor final do produto de software.

⊛ **Funções fuzzy:** mapeiam os atributos de qualidade primitivos ou agregados, através de termos lingüísticos, quantificando-os.

### III - USO DA TEORIA FUZZY EM QUALIDADE DE SOFTWARE

A teoria dos conjuntos *fuzzy*, em várias áreas do conhecimento humano, vem sendo o elo de ligação entre modelos imprecisos (subjetivos) do mundo real e sua representação matemática (DUBOIS, 1991).

Na ótica dessa teoria, cada atributo de qualidade de software pode ser visto como uma *variável lingüística*, relacionada a um conjunto de *termos lingüísticos*, associados a *funções de pertinência*, em um *conjunto referencial* estabelecido previamente. Cada atributo de qualidade será uma composição de termos lingüísticos, obtidos em um processo de avaliação.

Os termos lingüísticos  $T_i$ , para  $i = 1, 2, \dots, n$ , serão representados por *números fuzzy normais triangulares*  $\tilde{N}_i(a_i, m_i, b_i)$  do tipo-LR (LEE, 1996b, HSU, 1996, RÖMER, 1995, HAPKE, 1994, BARDOSSY, 1993, LASEK, 1992, RUONING, 1992, DUBOIS, 1980), que denotarão o grau de importância de cada atributo considerado, onde  $a_i < b_i$  e  $a_i \leq m_i$  ou  $m_i \leq b_i$ . O valor de  $n$  pode ser estabelecido de acordo com as conveniências do projeto, possíveis peculiaridades do domínio de aplicação ou determinação da equipe gestora da qualidade.

Na medição do nível de pressão sanguínea, por exemplo, pode ser suficiente apenas os três termos lingüísticos: *alta*, *normal* e *baixa*, para que se consiga diagnosticar um paciente apropriadamente. Já na aplicação de conceitos a alunos, muitas vezes são utilizados quatro termos lingüísticos: *excelente* (A), *bom* (B), *regular* (C), e *deficiente* (D). Entretanto, uma vez estabelecido o valor de  $n$  e o conjunto de termos lingüísticos para o domínio de aplicação, ou para o objeto a ser apreciado, isto deve permanecer constante ao longo de todo o processo de avaliação. Assim sendo, nas situações apresentadas acima, não se poderia usar, por exemplo, as expressões: *pressão deficiente* (a pressão estaria *alta* ou *baixa*?), nem *aluno normal* (o conceito desse aluno seria A, B ou C?).

Neste contexto, baseado em (HSU, 1996, LEE, 1996a, KACPRZYK, 1992, BALDWIN, 1979, ZADEH, 1977), sugere-se o conjunto de termos lingüísticos da Tabela 1, que poderá ser utilizados, na avaliação da qualidade de um produto de software.

Grau de importância	Simbologia	Termo Lingüístico	Número fuzzy normal
0,0	NR	Nenhuma Relevância	$\tilde{N}_1 = (0,0;0,0;1,0)$
1,0	PR	Pouca Relevância	$\tilde{N}_2 = (0,0;1,0;2,0)$
2,0	R	Relevante	$\tilde{N}_3 = (1,0;2,0;3,0)$
3,0	MR	Muito Relevante	$\tilde{N}_4 = (2,0;3,0;4,0)$
4,0	I	Imprescindível	$\tilde{N}_5 = (3,0;4,0;4,0)$

Tabela 1: Números *fuzzy* normais para termos lingüísticos

O conjunto dos termos lingüísticos da tabela acima possui as seguintes funções de pertinência, adaptadas de LEE (1996b), representadas na Figura 2.

$$\tilde{N}_1 = (0,0; 0,0; 1,0) \quad \mu_{N_1}(x) = \begin{cases} 1-x, & 0 \leq x \leq 1 \\ 0, & 1 \leq x \leq n \end{cases}$$

$$\tilde{N}_k = (k-2; k-1; k) \quad \mu_{N_k}(x) = \begin{cases} 0, & 0 \leq x \leq k-2 \\ x-(k-2), & k-2 \leq x \leq k-1 \\ k-x, & k-1 \leq x \leq k \\ 0, & k \leq x \leq n \end{cases} \quad \text{para } k = 2, 3, 4$$

$$\tilde{N}_n = (3,0; 4,0; 4,0) \quad \mu_{N_n}(x) = \begin{cases} 0, & 0 \leq x \leq 3 \\ x-3, & 3 \leq x \leq 4 \end{cases}$$

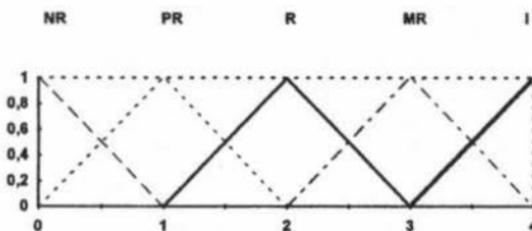


Figura 2: Funções de pertinência de números fuzzy, para termos linguísticos

Depois de estabelecido o conjunto de termos linguísticos e suas funções características correspondentes, a 'avaliação fuzzy', segundo o Modelo Rocha Estendido, desenvolvido em BELCHIOR (1997), será apresentada, a seguir.

#### IV - O MODELO FUZZY PARA AVALIAÇÃO DA QUALIDADE DE SOFTWARE

O Modelo fuzzy para avaliação da qualidade de software possui cinco etapas, para a consecução de seus objetivos, que pode envolver três situações distintas:

① **Determinação do Padrão de Qualidade (PQ) de um produto de software ou de um domínio de aplicação.** Procura-se obter de especialistas do produto (ou do domínio de aplicação) o grau de importância para cada atributo, de forma que o produto seja considerado de qualidade, tomando-se, como base, o objeto da avaliação. Isto significa dizer que o peso atribuído a cada atributo, por um especialista, deve retratar como o produto deveria estar. Portanto, neste caso, não se está avaliando o estado de um certo produto, mas o padrão ideal de qualidade que ele deveria apresentar.

② **Avaliação da qualidade de um produto de software, apoiando-se em um PQ já previamente definido.** Cada especialista julga o conjunto de atributos de qualidade, considerando o estado em que o software se encontra. Os resultados deste julgamento serão confrontados com o PQ, já estabelecido para o produto ou para o domínio de aplicação. São gerados, então, índices de qualidade, para cada atributo considerado, chegando-se à medição da qualidade do produto final. Esses índices medem o quanto o produto avaliado atinge, percentualmente, do padrão ideal de qualidade estabelecido, que tem índice igual a 1.

③ **Avaliação da qualidade de um produto de software, sem que haja um PQ já estabelecido.** Neste caso, como não há um PQ definido, os resultados serão apurados, levando-se em conta apenas o julgamento dos especialistas para o produto em questão. Com este procedimento, a equipe de desenvolvimento ou a equipe gestora da qualidade do produto terá um conjunto de dados úteis, que poderá auxiliar na continuidade do desenvolvimento do produto ou servir de parâmetros para possíveis melhorias em um produto já desenvolvido. A seguir, mostrar-se-á as etapas do modelo fuzzy para avaliação da qualidade de software.

**PRIMEIRA ETAPA: identificação do objeto a ser avaliado, do conjunto de atributos de qualidade de software a ser considerado na avaliação, e das instituições pesquisadas.**

1.1. **Estabelecer o objeto da avaliação:** nesta etapa, estabelece-se qual é o produto de software, que será objeto da avaliação. Podem ser avaliados subprodutos gerados

durante as diversas fases do ciclo de vida de um produto de software ou o próprio produto final.

1.2. **Definir o conjunto de atributos de qualidade:** o conjunto de atributos é definido considerando-se o objeto a ser avaliado, o domínio da aplicação e a tecnologia de desenvolvimento. Podem acontecer duas situações. Na primeira, os atributos já estarem definidos, fruto de trabalhos anteriores. Para o Modelo Rocha e, portanto, passíveis de utilização com o Modelo Rocha Estendido, tem-se conjuntos de atributos definidos para software científico (PALERMO, 1989), software financeiro (BELCHIOR, 1992), software educacional (CAMPOS, 1994), sistemas especialistas (OLIVEIRA, 1995), software médico (CARVALHO, 1994), sistemas de informação (BLASCHEK, 1995) e software orientado a objetos (CLUNIE, 1997). Na segunda situação, os atributos ainda não estão definidos e deve-se proceder à sua identificação, podendo-se, para isto, contar com a ferramenta proposta por (PASSOS, 1995).

1.3. **Selecionar a(s) instituição(ões) onde a pesquisa será realizada:**

- *Em uma única instituição:* quando esta processa a avaliação de atributos de qualidade de um produto de software, que lhe é próprio.
- *Em várias instituições:* quando dados são coletados com o objetivo de ser definido o padrão de qualidade de produtos de software, em uma área de aplicação específica.

#### SEGUNDA ETAPA: escolha dos especialistas.

2.1. **Obter o perfil dos especialistas:** obtenção do perfil dos especialistas (ISHIKAWA, 1993),  $E_i$  ( $i = 1, 2, \dots, n$ ), que participarão do processo de investigação, para se ter a indicação da importância relativa de cada um deles. De uma maneira geral, todos os indivíduos, que estão ou já estiveram, direta ou indiretamente, envolvidos com produtos similares ao objeto de avaliação poderão ser escolhidos. Esta etapa pode ser realizada através do *Questionário de Identificação do Perfil Especialista (QIPE)*, proposto em BELCHIOR (1997) que, neste caso, possui 7 itens ( $i_1, i_2, \dots, i_7$ ) para a avaliação de cada especialista, envolvendo, essencialmente, a experiência do mesmo no desenvolvimento de sistemas e seu grau de treinamento na área de informática.

2.2. **Calcular o peso do especialista:** calcula-se o grau de importância relativa de cada especialista, isto é, o peso  $p_{Ei}$ , através dos dados obtidos pelo preenchimento do *QIPE*, levando-se em consideração os seguintes critérios:

- cada *QIPE* contém as informações de um único especialista;
- O total de escores de cada especialista,  $tQIPE_i$ , é calculado, segundo indicações contidas na apuração de seus resultados (BELCHIOR, 1997).
- o peso de cada especialista  $p_{Ei}$ , que é sempre relativo em relação aos outros especialistas (média ponderada), será definido por:

$$p_{Ei} = \frac{tQIPE_i}{\sum_{i=1}^n tQIPE_i}$$

#### TERCEIRA ETAPA: determinação do grau de importância de cada atributo de qualidade, identificado na Etapa 1.

O processo de investigação consiste em se obter dos especialistas selecionados, graus de importância, para se obter o julgamento dos mesmos com relação a cada um dos atributos de qualidade mensuráveis (critérios), através da utilização do conjunto de

termos lingüísticos, caracterizados por números *fuzzy*,  $\tilde{N}_i(a_i, m_i, b_i)$ , previamente delineados. Caso o especialista deseje utilizar um outro número *fuzzy*  $\tilde{N}_k(a_k, m_k, b_k)$ , que não pertença ao conjunto previamente estabelecido, poderá fazê-lo, desde que  $k \in [0, n]$  e  $\tilde{N}_i < \tilde{N}_k < \tilde{N}_n$ . Seja, por exemplo, o número *fuzzy*  $\tilde{N}_k = (1,5; 2,4; 3,6)$ , utilizado por um especialista, teria significado semântico no intervalo entre os termos lingüísticos *relevante* e  *muito revelaste* da Tabela 1.

O especialista deverá ter sido comunicado de que o processo de investigação está sendo aplicado, para se levantar um determinado Padrão de Qualidade (PQ), ou apenas se estar realizando uma avaliação do estado de um produto de software, para que ele possa fazer seus julgamentos coerentemente. Serão realizadas as seguintes atividades:

- 3.1. **Definir o procedimento de investigação:** esse procedimento poderá consistir na elaboração de um *questionário* ou um outro dispositivo de investigação, e na definição de técnicas de aplicação próprias para o mesmo, usando-se *graus de importância* (dos termos lingüísticos) estabelecidos.
- 3.2. **Aplicar o dispositivo de investigação aos especialistas:** o dispositivo de investigação é aplicado aos especialistas selecionados na etapa anterior.

#### QUARTA ETAPA: tratamento dos dados coletados dos especialistas, na avaliação de cada atributo de qualidade mensurável considerado (critério).

Nesta etapa, os prognósticos individuais obtidos de cada especialista são combinados, para cada um dos atributos de qualidade diretamente mensuráveis (critérios), gerando-se uma opinião consensual entre esses especialistas, para cada critério avaliado, sendo formalizada por uma função de pertinência *fuzzy* característica  $\tilde{N}$ , onde:

$$\tilde{N} = f(\tilde{N}_1, \tilde{N}_2, \dots, \tilde{N}_n)$$

- 4.1. **Calcular o grau de concordância** (HSU, 1996, CHEN, 1993, ZWICK, 1987): calcula-se o grau de concordância,  $C(\tilde{N}_i, \tilde{N}_j)$ , combinando-se os julgamentos dos especialistas  $E_i$  e  $E_j$ , através da razão entre a área de interseção entre eles e a área total, de suas funções de pertinência, isto é:

$$C(\tilde{N}_i, \tilde{N}_j) = \frac{\int_x (\min\{\mu_{\tilde{N}_i}(x), \mu_{\tilde{N}_j}(x)\}) dx}{\int_x (\max\{\mu_{\tilde{N}_i}(x), \mu_{\tilde{N}_j}(x)\}) dx}$$

- 4.2. **Construir a matriz de concordância:** calculados todos os graus de concordância entre cada par de especialistas  $E_i$  e  $E_j$ , constrói-se uma *matriz de concordância*, MC, que fornece as indicações da aquiescência entre eles, onde  $C_{ij} = C(\tilde{N}_i, \tilde{N}_j)$ , se  $i \neq j$  e  $C_{ij} = 1$ , se  $i = j$ .

$$MC = \begin{bmatrix} 1 & C_{12} & \dots & C_{1j} & \dots & C_{1n} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ C_{n1} & C_{n2} & \dots & C_{nj} & \dots & C_{nn} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ C_{n1} & C_{n2} & \dots & C_{nj} & \dots & 1 \end{bmatrix}$$

Após a construção da matriz MC, deve-se observar os seguintes itens:

- i. Caso  $C_{ij} = 0$ , isto é, não há interseção entre o  $i$ -ésimo e o  $j$ -ésimo especialista, obtém-se mais informações desses especialistas (segundo a conveniência da avaliação), com o objetivo de ajustar (convergir) suas opiniões, isto é, chegar a

uma interseção entre eles.

- ii. Depois de obtidas as novas informações, conforme o item (i), se persistir algum grau de concordância nulo, considerá-lo na matriz  $MC$ , pois, no processo de agregação (item 4.6), os valores nulos (isto é, os que destoaram do consenso geral entre os especialistas) terão peso zero no resultado final da agregação.
- iii. Atentar para o fato de que se houver uma grande disparidade de respostas (isto é, um baixo consenso entre os especialistas), isto pode significar que estes não entenderam convenientemente a definição do objeto de investigação (DYER, 1992). Neste caso, o item (i) deve ser executado, tanto quanto possível, no sentido de se chegar ao maior consenso entre esses especialistas.
- 4.3. **Calcular a concordância relativa:** através dos dados obtidos de  $MC$ , calcula-se a *concordância relativa* ( $CR_i$ ) de cada especialista envolvido neste processo, pela *média quadrática* do grau de concordância entre eles:

$$CR_i = \sqrt{\frac{1}{n-1} \sum_{j=1}^n C_{ij}^2}$$

Com este procedimento, o cálculo de  $CR_i$  tenderá para os maiores índices de consenso entre os especialistas avaliadores.

- 4.4. **Calcular o grau de concordância relativa:** o grau da concordância *relativa* ( $GCR_i$ ) de um especialista, em relação a todos os outros especialistas, é obtido pela *média ponderada* de  $CR_i$  de cada especialista:

$$GCR_i = \frac{CR_i}{\sum_{i=1}^n CR_i}$$

- 4.5. **Calcular o coeficiente de consenso dos especialistas:** o *coeficiente de consenso*, obtido para cada especialista ( $CCE_i$ ) considera tanto o  $GCR_i$ , quanto o peso,  $p_{Ei}$  (item 2.2), de cada um deles.

$$CCE_i = \frac{GCR_i \cdot p_{Ei}}{\sum_{i=1}^n (GCR_i \cdot p_{Ei})}$$

- 4.6. **Avaliar o critério de qualidade:** o resultado da avaliação de cada critério de qualidade será dado por  $\tilde{N}$ , que também é um número *fuzzy* normal triangular, onde  $\bullet$  é o *produto algébrico fuzzy* (HSU, 1996, KAUFMANN, 1991):

$$\tilde{N} = \sum_{i=1}^n (CCE_i \bullet \tilde{N}_i)$$

#### QUINTA ETAPA: agregação dos atributos de qualidade de software, em cada nível hierárquico do modelo de qualidade.

Nesta etapa, faz-se a agregação dos atributos de qualidade do tipo  $\tilde{N}$ , de acordo com o Modelo Rocha Estendido, gerando-se uma função de pertinência *fuzzy* característica, para cada subconjunto de atributos de qualidade em questão, isto é, os *atributos agregados*. Assim sendo, tem-se de 1 a  $o$  *objetivos* de qualidade para um produto de software. Para cada objetivo, são definidos de 1 a  $f$  *fatores* de qualidade a ele relacionados. Cada *fator* contém de 0 a  $s$  *subfatores*. Cada *fator/subfator* possui de 1 a  $c$  *critérios*. Cada *atributo agregado* avaliado,  $\tilde{N}$ , composto por seus atributos constituintes,  $\tilde{N}_1, \tilde{N}_2, \dots, \tilde{N}_n$ , também será formalizado por:

$$\tilde{N} = f(\tilde{N}_1, \tilde{N}_2, \dots, \tilde{N}_n)$$

5.1 **Estabelecer o padrão de qualidade:** quando se tratar do estabelecimento do padrão de qualidade ( $PQ$ ) para um determinado produto de software ou para um domínio de aplicação específico, calcula-se o peso  $W_i$  (BELCHIOR, 1996a, BELCHIOR, 1996b, CHEN, 1995, LIOU, 1992), isto é, o grau de contribuição de cada atributo, que compõe o atributo agregado avaliado. O peso de cada atributo  $W_i$  será obtido pela média ponderada dos graus de importância de cada um de seus atributos constituintes, isto é, o valor  $w_i$ , que será calculado através da defuzzificação de seu número fuzzy  $\tilde{N}_i(a_i, m_i, b_i)$  correspondente (transformação de um número fuzzy em um número real). Portanto:

i.  $w_i = m_i$ , que corresponde ao valor do grau de pertinência igual a 1, isto é, o valor (número real) do atributo de qualidade.

ii.  $W_i = w_i / \sum w_i$

5.2 **Calcular o grau de concordância agregado:** calcula-se o grau de concordância,  $C(\tilde{N}_i, \tilde{N}_j)$ , entre os atributos de qualidade, que estão sendo agregados (que são números fuzzy  $\tilde{N}$ ) da mesma maneira que no item 4.1.

5.3 **Construir a matriz de concordância de agregação:** após o cálculo de todos os  $C_{ij}$  dos atributos pertencentes ao subconjunto, que está sendo agregado, gera-se a matriz de concordância de agregação ( $MCA$ ) similarmente ao item 4.2. Se  $C_{ij} = 0$ , não há interseção entre os atributos  $i$  e  $j$  e, neste caso, procede-se ao cálculo do grau de não concordância,  $\bar{C}_{ij}$ , entre esses atributos, que deverá ser um valor no intervalo  $[-1, 0]$ :

$$\bar{C}_{ij} = -\frac{d}{D} \cdot r, \text{ onde:}$$

i.  $d$  é a menor distância entre os dois números fuzzy considerados, ou seja,  $d = a_j - b_i$  ou  $d = a_i - b_j$  (utiliza-se o menor valor absoluto de  $d$ ).

ii.  $D$  é a maior distância entre o maior e o menor termo lingüístico dos termos lingüísticos considerados, isto é,  $\tilde{N}_n$  e  $\tilde{N}_1$  respectivamente. Portanto,  $D = a_n - b_1$ .

iii.  $r$  é a razão entre as áreas dos números fuzzy  $\tilde{N}_i$  e  $\tilde{N}_j$ , sendo que  $0 < r \leq 1$ , daí:

$$r = \frac{\int_x (\mu_{\tilde{N}_i}(x)) dx}{\int_x (\mu_{\tilde{N}_j}(x)) dx} \text{ ou } r = \frac{\int_x (\mu_{\tilde{N}_j}(x)) dx}{\int_x (\mu_{\tilde{N}_i}(x)) dx}$$

5.4 **Construir a matriz de agregação:** constrói-se a nova matriz de agregação,  $MA$ , que contem os valores de  $C_{ij}$  e  $\bar{C}_{ij}$ , isto é, os grau dos estados de agregação,  $E_{ij}$ , substituindo a matriz  $MCA$ .

5.5 **Calcular o estado relativo de agregação:** obtém-se o estado relativo de agregação,  $ERA$  (era), através da concordância ou não concordância de cada um dos atributos, que estão sendo agregados, através de dois procedimentos:

i. Quando existe algum grau de não concordância,  $\bar{C}_{ij}$ , na matriz  $MA$ , calcula-se, inicialmente, o 'era' de cada atributo, pela média aritmética do grau de concordância e do grau de não concordância do atributo em questão com os outros, que estão sendo agregados, quantificando-se, assim, o quanto o atributo concorda ou diverge em relação aos demais, no processo de agregação.

$$era_i = \frac{1}{n-1} \sum_{j=1}^n E_{ij}$$

Com este procedimento, visualiza-se o estado de cada atributo no processo de agregação, isto é, o quanto cada atributo realmente contribui na composição do novo atributo agregado. Quando algum atributo agregante tiver seu 'era' não positivo, pode-se inquirir se, de fato, esse atributo deveria pertencer ou não àquele ramo da hierarquia de composição dos atributos do produto de software que está sendo avaliado. Esta pode ser uma informação útil, para a validação da árvore hierárquica dos atributos de qualidade de software.

- ii. Uma vez que os atributos, que estão sendo agregados, realmente pertencem ao ramo hierárquico da árvore dos atributos de qualidade em questão, procede-se ao cálculo do 'ERA' pela média quadrática dos graus dos estados de agregação,  $E_{ij}$ , de seus atributos agregantes.

$$ERA_i = \sqrt{\frac{1}{n} \sum_{j=1}^n E_{ij}^2}$$

Neste caso, os graus negativos (de não concordância) são tratados e exercem a mesma influência como se fossem positivos. Com este procedimento, objetiva-se chegar a um valor compensatório da contribuição de cada atributo no processo de agregação, através do modelo fuzzy proposto. Desta forma, preservando-se as características particulares de composição dos atributos, isto é, sua localização na árvore hierárquica de atributos de qualidade estabelecida pela equipe gestora da qualidade na organização, executa-se o processo de agregação.

Exemplificando: se um determinado atributo de qualidade de software for composto por dois outros atributos com graus de importância diferentes entre si, para um determinado produto e, se um desses atributos tiver um grau de importância *baixo* e o outro *alto*, isto significa que eles contribuiriam em uma percentagem baixa e alta, respectivamente, para a composição do atributo agregado considerado. Através do procedimento (ii) acima, obter-se-ia para o atributo agregado um grau de importância *médio*. Portanto, o atributo agregado avaliado tem grau de importância *médio*, sendo composto por dois atributos, que possuem graus de importância *baixo* e *alto*.

No entanto, se na análise dos resultados, este processo de agregação não for conveniente para o produto que está sendo avaliado, deve-se justificar o motivo da agregação não poder ser realizada como tal, relacionando-se apenas os graus de importância de cada atributo, que compõem o atributo, que seria agregado.

- 5.6 **Calcular o grau do estado relativo de agregação:** o grau do estado relativo de agregação,  $GERA_i$ , de cada atributo agregado, é obtido pela média ponderada entre seus atributos constituintes:

$$GERA_i = \frac{ERA_i}{\sum_{i=1}^n ERA_i}$$

- 5.7 **Calcular o coeficiente de consenso do atributo:** o coeficiente de consenso do atributo ( $CCA_i$ ), obtido para cada atributo, que compõe o atributo que está sendo agregado, considerará tanto o  $GERA_i$ , como também o peso  $W_i$  (item 5.1 (ii)) de cada um desses atributos. Caso não haja ainda um padrão de qualidade ( $PQ$ ) estabelecido para o produto de software, que está sendo avaliado, ou para o seu domínio de aplicação,

deve-se considerar  $W_i = 1$ , isto é,  $CCA_i = GERA_i$  (item 5.6).

$$CCA_i = \frac{GERA_i \cdot W_i}{\sum_{i=1}^n (GERA_i \cdot W_i)}$$

5.8 **Avaliar o atributo agregado:** o resultado da avaliação de cada atributo de qualidade agregado será dado por  $\tilde{N}$ , que também é um número *fuzzy*, onde  $\bullet$  é o *produto algébrico fuzzy* (KAUFMANN, 1991), formalizado por:

$$\tilde{N} = \sum_{i=1}^n (CCA_i \bullet \tilde{N}_i)$$

Baseando-se nos resultados obtidos deste modelo *fuzzy* para avaliação da qualidade de software, pode-se definir índices de qualidade, na avaliação de novos produtos de software, de acordo com o padrão de qualidade estabelecido.

## V - DEFINIÇÃO DE ÍNDICES DE QUALIDADE DE SOFTWARE

Se houver um *padrão de qualidade (PQ)* já definido para o produto de software, que está sendo avaliado ou seu domínio de aplicação, pode-se comparar os resultados obtidos pela aplicação do modelo *fuzzy* com esse padrão. Com isto, obtém-se um *índice de qualidade*, que indicará se o produto de software avaliado está ou não dentro do padrão de qualidade desejado, e em que percentagem. Assim sendo, pode-se determinar o *índice de qualidade* do produto avaliado, através das seguintes ações:

- **Redefinição da função característica do atributo de qualidade:**

Redefine-se a função *fuzzy* de cada atributo de qualidade do tipo  $\tilde{N}_i = (a_i, m_i, b_i)$ , obtida na definição do *padrão de qualidade*, para a *função fuzzy do padrão de qualidade*,  $\tilde{Q}_i = (a_i, \underline{m}_i, \bar{m}_i, b_i)$ , isto é, um número *fuzzy* normal trapezoidal.

O mapeamento de  $\tilde{N}_i$  para  $\tilde{Q}_i$  resultará, então, na função  $\tilde{Q}_i = (a_i, m_i, b_n, b_n)$ , onde  $n$  é o limite superior do *conjunto referencial* já definido. Esse mapeamento é possível, porque pode-se considerar que qualquer valor acima do padrão de qualidade (à direita da função característica), é também de qualidade e, portanto, plenamente aceitável.

- **Cálculo do índice de qualidade:**

O *índice de qualidade*,  $q_k$ , para cada atributo  $k$ , que está sendo avaliado, é dado por:

$$q_k = \frac{\int_x (\min\{\mu_{\tilde{Q}_i}(x), \mu_{\tilde{N}_i}(x)\}) dx}{\int_x (\mu_{\tilde{N}_i}(x)) dx}$$

Uma vez que  $q \in [0, 1]$ , quando  $q = 1$ , isto significa que o atributo avaliado atinge totalmente o padrão de qualidade; se  $q = 0$ , então o elemento avaliado está totalmente fora do padrão de qualidade; se  $0 \leq q \leq 1$ , o atributo está dentro do padrão de qualidade, na proporção do valor de  $q$ , isto é,  $q\%$  do padrão de qualidade.

No cálculo do *índice de qualidade* para atributos agregados, se algum de seus atributos constituintes tiver  $q = 1$ , considerar, neste caso, o valor do *PQ* desse atributo constituinte.

## VI - VALIDAÇÃO DO MODELO

O Modelo *Fuzzy* para Avaliação da Qualidade de Software foi validado considerando-se duas das situações previstas para a sua utilização. O objeto da avaliação foi Especificações de Requisitos de Software (ERS) em geral, sendo, portanto, realizada:

a) *A determinação do Padrão de Qualidade (PQ) de um produto de software*: 16 especialistas, de 3 instituições com larga experiência no desenvolvimento de produtos de software, responderam a uma pesquisa de campo realizada por CLUNIE (1997). O objetivo dessa pesquisa foi obter, de cada especialista selecionado, o grau de importância para cada um dos atributos de ERS, definidos, também, em CLUNIE (1997), para que essa ERS fosse considerada de qualidade, isto é, para se obter seu *padrão de qualidade*.

b) *A avaliação da qualidade de um produto de software, apoiada no PQ estabelecido*: um grupo de 3 especialistas avaliou a modelagem de dados do SIGAH-Multimídia, um sistema de informação hospitalar, sendo considerado apenas um subconjunto dos atributos levantados no item anterior, por conter os atributos passíveis de avaliação para essa modelagem (subproduto da ERS), sob avaliação. Os resultados deste julgamento foram confrontados com o PQ estabelecido para ERS, sendo gerados *índices de qualidade* para cada atributo avaliado.

Desta forma, foram validados todas as etapas previstas no modelo, que se comportou conforme o previsto. Por razão de espaço é impossível reproduzir, neste artigo, toda a experiência feita de utilização desse modelo *fuzzy*. Portanto, serão reproduzidos, apenas, os resultados finais de avaliação do item b. A Tabela 2 mostra o resultado do QIPE (Questionário de Identificação do Perfil do Especialista).

Apuração dos Resultados do QIPE									
$E_i / \text{item}$	$I_1$	$I_2$	$I_3$	$I_4$	$I_5$	$I_6$	$I_7$	IQIPE	$P_B$
1	2,80	1,30	2,40	0,30	0,70	0,80	5,90	4,177	0,260
2	2,70	6,00	2,50	0,70	1,00	0,80	9,00	5,990	0,373
3	3,10	3,00	3,90	0,70	0,90	0,80	9,20	5,900	0,367
Total								Total	Total
3								16,067	1,000

Tabela 2: Resultados do QIPE

O conjunto de termos lingüísticos utilizado no *dispositivo de investigação* (questionário), para avaliação do subproduto da ERS do SIAGH-Multimídia, possui uma relação direta com o conjunto de termos lingüísticos estabelecido no levantamento do padrão de Qualidade (PQ) para ERS, como é exibido na Tabela 3.

Grau de Importância	Termo Lingüístico do PQ	Termo Lingüístico SIAGH	Número <i>fuzzy</i> normal
0,0	Nenhuma Relevância	Total Ausência	$\tilde{N}_1 = (0,0; 0,0; 1,0)$
1,0	Pouca Relevância	Baixa Presença	$\tilde{N}_2 = (0,0; 1,0; 2,0)$
2,0	Relevante	Moderada Presença	$\tilde{N}_3 = (1,0; 2,0; 3,0)$
3,0	Muito Relevante	Alta Presença	$\tilde{N}_4 = (2,0; 3,0; 4,0)$
4,0	Imprescindível	Total presença	$\tilde{N}_5 = (3,0; 4,0; 4,0)$

Tabela 3: Relação entre os números *fuzzy* normais do PQ e do SIAGH-Multimídia

A técnica de avaliação utilizada foi a *inspeção individual*, tendo como base o documento da modelagem de dados do SIAGH-Multimídia, e o subconjunto dos requisitos selecionados para esta avaliação, que pode ser identificado na Tabela 4.

ATRIBUTOS DE QUALIDADE DE SOFTWARE	PQ para ERS	ERS	IQ
<b>OBJETIVO: CONFIABILIDADE DA REPRESENTAÇÃO</b>	<i>N</i> (2,48; 3,48; 3,93)	---	
<b>FATOR: COMUNICABILIDADE (COB)</b>	<i>N</i> (2,39; 3,39; 3,91)	---	
Subfator: Correção no Uso do Método (CUM)	<i>N</i> (2,47; 3,47; 3,94)	<i>N</i> (3,00; 4,00; 4,00)	1,000
Critério: Correção da Notação (CNT)	<i>N</i> (2,55; 3,55; 3,99)	<i>N</i> (3,00; 3,00; 4,00)	1,000
Correção Sintática (CST)	<i>N</i> (2,40; 3,40; 3,95)	<i>N</i> (3,00; 3,00; 4,00)	1,000
Correção Semântica (CSM)	<i>N</i> (2,86; 3,86; 4,00)	<i>N</i> (3,00; 4,00; 4,00)	1,000
Correção no Uso do Formato de Document. (CFD)	<i>N</i> (2,06; 3,06; 3,82)	<i>N</i> (3,00; 4,00; 4,00)	1,000
Subfator: Uniformidade de Terminologia (UTE)	<i>N</i> (2,66; 3,66; 4,00)	<i>N</i> (2,17; 3,17; 3,75)	0,477
Uniformidade de Termos (UTT)	<i>N</i> (2,74; 3,74; 4,00)	<i>N</i> (2,17; 3,17; 3,51)	0,328
Uniformidade de Notação (UNT)	<i>N</i> (2,57; 3,57; 4,00)	<i>N</i> (2,17; 3,17; 4,00)	0,607
Subfator: Uniformidade no Nível de Abstração (UNA)	<i>N</i> (1,55; 2,55; 3,48)	<i>N</i> (2,47; 3,47; 4,00)	1,000
Uniformidade de Detalhes da Documentação (UDD)	<i>N</i> (1,72; 2,72; 3,65)	<i>N</i> (2,00; 3,00; 4,00)	1,000
Independência de Detalhes do Projeto (IDP)	<i>N</i> (1,37; 2,37; 3,31)	<i>N</i> (3,00; 4,00; 4,00)	1,000
Subfator: Modularidade da Documentação (MDO)	<i>N</i> (2,36; 3,36; 3,97)	---	
Coesão de Informações (COI)	<i>N</i> (2,35; 3,35; 4,00)	---	
Acoplamento entre as Seções (ACS)	<i>N</i> (2,10; 3,10; 3,91)	---	
Estrutura da Documentação (EDO)	<i>N</i> (2,64; 3,64; 3,99)	---	
Subfator: Concisão (COC)	<i>N</i> (2,55; 3,55; 3,86)	<i>N</i> (1,65; 2,65; 3,27)	0,199
Complementabilidade (COM)	<i>N</i> (2,55; 3,55; 3,86)	<i>N</i> (1,65; 2,65; 3,27)	0,199
Subfator: Conformidade (COF)	<i>N</i> (2,41; 3,41; 3,97)	---	
Aderência às Normas Org. Desenvol. (ANO)	<i>N</i> (2,22; 3,22; 3,95)	---	
Aderência às Normas estab. p/ Contratante (ANC)	<i>N</i> (2,61; 3,61; 3,99)	---	
<b>FATOR: MANIPULABILIDADE (MAP)</b>	<i>N</i> (2,75; 3,75; 3,98)	---	
Subfator: Disponibilidade (DIS)	<i>N</i> (2,95; 3,95; 4,00)	<i>N</i> (2,48; 3,48; 3,93)	
Acessibilidade (ACE)	<i>N</i> (2,90; 3,90; 4,00)	<i>N</i> (2,32; 3,32; 3,66)	0,000
Estar Atualizada (ETA)	<i>N</i> (3,00; 4,00; 4,00)	<i>N</i> (1,04; 2,04; 2,70)	0,000
Subfator: Rastreabilidade (RAS)	<i>N</i> (2,56; 3,56; 3,96)	---	
Organização da Documentação (ORD)	<i>N</i> (2,88; 3,88; 4,00)	---	
Localizabilidade Interna (LOI)	<i>N</i> (2,56; 3,56; 4,00)	<i>N</i> (0,00; 1,00; 1,00)	0,000
Localizabilidade Externa (LOE)	<i>N</i> (2,23; 3,23; 3,87)	---	
<b>OBJETIVO: CONFIABILIDADE CONCEITUAL</b>	<i>N</i> (2,49; 3,49; 3,90)	---	
<b>FATOR: FIDELIDADE (FID)</b>	<i>N</i> (2,69; 3,69; 3,97)	---	
Subfator: Consistência (CON)	<i>N</i> (2,67; 3,67; 3,94)	<i>N</i> (2,90; 3,90; 4,00)	0,961
Critério: Consistência Interna (COI)	<i>N</i> (2,89; 3,89; 4,00)	<i>N</i> (2,82; 3,82; 4,00)	0,876
Consistência Externa (COE)	<i>N</i> (2,45; 3,45; 3,88)	<i>N</i> (3,00; 4,00; 4,00)	1,000
Subfator: Não Ambiguidade (NAB)	<i>N</i> (2,70; 3,70; 3,99)	<i>N</i> (2,71; 3,71; 4,00)	0,871
Ser Explícita EXP)	<i>N</i> (2,81; 3,81; 3,99)	<i>N</i> (3,00; 4,00; 4,00)	1,000
Precisão (PRC)	<i>N</i> (2,60; 3,60; 3,99)	<i>N</i> (2,40; 3,40; 4,00)	0,770
<b>FATOR: SUFICIÊNCIA (SUF)</b>	<i>N</i> (2,23; 3,23; 3,81)	<i>N</i> (2,88; 3,88; 4,00)	1,000
Subfator: Necessidade (NEC)	<i>N</i> (2,92; 3,92; 4,00)	<i>N</i> (3,00; 4,00; 4,00)	1,000
Necessidade dos Requisitos (NRQ)	<i>N</i> (2,92; 3,92; 4,00)	<i>N</i> (3,00; 4,00; 4,00)	1,000
Subfator: Não Redundância (NRD)	<i>N</i> (1,84; 2,84; 3,61)	<i>N</i> (2,82; 3,82; 4,00)	1,000
Não Redundância de Informações (NRI)	<i>N</i> (1,84; 2,84; 3,61)	<i>N</i> (2,82; 3,82; 4,00)	1,000
Subfator: Completude (COP)	<i>N</i> (2,18; 3,18; 3,83)	---	
Completude Relação Roteiro def. p/ Org. (CRO)	<i>N</i> (2,28; 3,28; 3,89)	---	
Completude e/Relação Método Desenv. (CMM)	<i>N</i> (1,63; 2,63; 3,58)	<i>N</i> (2,00; 3,00; 4,00)	1,000
Completude com Relação aos Requisitos (COR)	<i>N</i> (2,57; 3,57; 4,00)	---	

Tabela 4: Resultados da avaliação para a modelagem de dados do SIAGH-Multimídia

A partir dos resultados apresentados na Tabela 4, pode-se inferir que o produto avaliado está dentro do *padrão de qualidade (PQ)* para ERS, segundo o *Modelo Fuzzy* proposto. Neste caso, dos atributos avaliados pelos especialistas, 58% estão com  $q = 1$ , isto é, com qualidade ideal, 26% dos critérios estão dentro do *PQ*, e apenas 16% estão abaixo desse padrão.

Os atributos que não atingiram a qualidade ideal ( $q = 1$ ), mas que estão dentro do padrão de qualidade, em termos percentuais, são: (i) *Uniformidade de Termos* - 32,8%; (ii) *Uniformidade de Notação* - 60,7%; (iii) *Complementabilidade* - 19,9%; (iv) *Consistência Interna* - 87,6%; e (v) *Precisão* - 77,0%.

## VII - CONCLUSÃO

O Modelo *Fuzzy* para Avaliação da Qualidade de Software, possui algumas características relevantes:

- i. **Preservação da concordância** (HSU, 1996, BARDOSSY, 1993): se todas as estimativas forem idênticas, a combinação resultante será a estimativa comum entre elas, isto é, se  $\tilde{N}_i = \tilde{N}_j$  para todo  $i, j$ , então  $\tilde{N} = \tilde{N}_i$ .
- ii. **Independência da ordem de combinação** (BARDOSSY, 1993) o resultado da agregação não depende da ordem da combinação das opiniões individuais. Se  $\{\sigma(1), \sigma(2), \dots, \sigma(n)\}$  é uma permutação de  $\{1, 2, \dots, n\}$ , então  $\tilde{N} = f(\tilde{N}_1, \tilde{N}_2, \dots, \tilde{N}_n) = f(\tilde{N}_{\sigma(1)}, \tilde{N}_{\sigma(2)}, \dots, \tilde{N}_{\sigma(n)})$ .
- iii. **Influência do grau de concordância e do peso do especialista**: se a estimativa de um especialista distancia-se das estimativas dos demais (pequeno índice de concordância), e/ou o peso do perfil desse especialista é inferior aos dos demais, então sua estimativa terá uma importância reduzida, no processo de avaliação.
- iv. **Preservação do número fuzzy**: se as opiniões *fuzzy* de todos os especialistas são representadas por um número *fuzzy* triangular positivo normal, então a função de pertinência da combinação é também um número *fuzzy* triangular positivo normal.

A validação do *Modelo fuzzy para avaliação da qualidade de software* foi, plenamente, satisfatória pois o modelo atendeu, convenientemente, a seus objetivos nesta experiência: (i) estabeleceu um padrão de qualidade para o produto de software considerado, e (ii) avaliou um produto de software (similar) real, com base no padrão estabelecido.

Com esse Modelo obteve-se uma base matemática sólida, fornecida pela teoria dos conjuntos *fuzzy*, e um mecanismo capaz de interpretar, na linguagem do desenvolvedor e/ou usuário, os resultados da medição de um produto (ou subproduto) de software.

## Referências Bibliográficas

- BALDWIN, J. F., 1979, *A new approach to approximate reasoning using a fuzzy logic*, Fuzzy Sets and Systems, 2,309-325.
- BARDOSSY, A., DUCKSTEIN, L., BOGARDI, I., 1993, *Combination of fuzzy number representing expert opinions*, Fuzzy Sets and Systems 57 (1993) 173-181.
- BELCHIOR, A. D., 1992, *Controle da Qualidade de Software Financeiro*, Tese de Mestrado, COPPE/UFRJ, Rio de Janeiro.
- BELCHIOR, A. D., XEXÉO, G. B., ROCHA, A. R. C., 1996a, *Aplicação da Teoria Fuzzy em Requisitos de Qualidade de Software*, XX II CLEI, Bogotá de Santa Fé, Colômbia.
- BELCHIOR, A. D., XEXÉO, G. B., ROCHA, A. R. C., 1996b, *Evaluating Software Quality Requirements using Fuzzy Theory*, International Conference on Information Systems Analysis and Synthesis (ISAS), Orlando, USA.
- BELCHIOR, A. D., 1997, *Um Modelo Fuzzy para Avaliação da Qualidade de Software*, Tese de Doutorado, COPPE/UFRJ, Rio de Janeiro, Maio.
- BLASCHECK, J. R., 1995, *Planejamento de Projetos*, Tese de Doutorado, COPPE/UFRJ.
- BOLOIX, G., et al., 1995, *A Software System Evaluation Framework*, IEEE Software, December.

CAMPOS, G. H. B., 1994, *Metodologia para avaliação da qualidade de software educacional: Diretrizes para desenvolvedores e usuários*, Tese de Doutorado, COPPE/ UFRJ.

CARVALHO, D., 1994, *Requisitos de Qualidade para o Software Médico*, Relatório Técnico, Fundação Bahiana de Cardiologia.

CHEN, C. T., HSY, H. M., 1993, *A study of fuzzy TOPSIS model*, Proc. of the Chinese Institute of Industrial Engineers National Conference, in (HSU, 1996).

CHEN, J. E., OTTO, K. N., 1995, *Constructing membership function using interpolation and measurement theory*, Fuzzy Sets and Systems 73, 313-327.

CLUNIE, C. E., 1997, *Avaliação da Qualidade de Especificações Orientadas a Objeto*, Tese de Doutorado, COPPE/UFRJ.

DUBOIS, D., PRADE, H., 1980, *Fuzzy Sets and Systems: Theory and Applications*, Academic Press, New York.

DUBOIS, D., PRADE, H., 1991, *Fuzzy sets in approximate reasoning. Part 1: Inference with possibility distributions*, Fuzzy Sets and Systems, IFSA, Special Memorial Volume: 25 years of fuzzy sets, North-Holland - Amsterdam.

DYER, M., 1992, *The Cleanroom Approach to Quality Software Development*, John Wiley & Sons, Inc. New York, 1992.

FENTON, N. E., PFLEEGER, S. L., 1997, *Software Metrics: A Rigorous & Practical Approach*, Second Edition, PWS Publishing Company.

HAPKE, M. et al., 1994, *Fuzzy project scheduling system for software development*, Fuzzy Sets and Systems 67, 101-117.

HSU, H. M., CHEN, C. T., 1996, *Aggregation of fuzzy opinions under group decision making*, Fuzzy Sets and Systems 79, 279-285.

IBRAHIM, A., AYYUB, B. M., 1992, *Multi-criteria ranking of components according to their priority for inspection*, Fuzzy Sets and Systems 48, 1-14.

ISHIKAWA, A. et al., 1993, *The max-min Delphi method and fuzzy Delphi method via fuzzy integration*, Fuzzy Sets and Systems 55, 241-253.

KACPRZYK, J. et al., 1992, *Group decision making and consensus under fuzzy preference and fuzzy majority*, Fuzzy Sets and Systems 49, 21-31.

KAUFMANN, A., GUPTA, M. M., 1991, *Introduction to Fuzzy Arithmetic: Theory and Applications*, Van Nostrand Reinhold, New York.

KITCHENHAM, B. et al., 1996, *Software Quality: The Elusive Target*, IEEE Software, January.

LASEK, M., 1992, *Hierarchical structures of fuzzy ratings in the analysis of strategic goal of enterprises*, Fuzzy Sets and Systems 50, 127-134.

LEE, H. M., 1996a, *Applying fuzzy set theory to evaluate the rate of aggregative risk in software development*, Fuzzy Sets and Systems 79, 323-336.

LEE, H. M., 1996b, *Group decision making using fuzzy theory for evaluating the rate of aggregative risk in software development*, Fuzzy Sets and Systems 80, 261-271.

LIU, T. S., JIUN, M., WANG, J., 1992 *Fuzzy weighted average: An improved algorithm*, Fuzzy Sets and Systems 49, 307-315.

MÖLLER, K. H., 1993, *Software Metrics: a practitioner's guide to improved product development*, Chapman & Hall Computing.

OLIVEIRA, K. M., 1995, *Avaliação da Qualidade de Sistemas Especialistas*, Tese de mestrado, COPPE/UFRJ, Rio de Janeiro, março.

PALERMO, S., ROCHA, A. R. C., 1989, *An experience on Evaluating Software Quality for High energy Physics*, Computer Physics Communications.

PASSOS, M. C. J. F., 1995, *Especificador da Meta de Qualidade de Projetos; Exame de qualificação*, Vitória.

PRATHER, R. E., 1996, *Convexity and independence in software metric theory*, Software Engineering Journal, July.

RIBEIRO, R. A., 1996, *Fuzzy multiple attribute decision making: A review and new preference elicitation techniques*, Fuzzy Sets and Systems 78, 155-181.

ROCHA, A. R. C., 1983, *Um Modelo para Avaliação da Qualidade de Especificações*. Tese de Doutorado, PUC-RJ, Rio de Janeiro.

RÖMER, C., KANDEL, A., 1995, *Statistical tests for fuzzy data*, FSS 72, 1-26.

RUONING, X., XIAOYAN, Z., 1992, *Extensions of the analytic hierachy process n fuzzy environment*, Fuzzy Sets and Systems 52, 251-257.

SCHNEIDEWIND, N. F., 1992, *Methodology for validating software metrics*, IEEE Transaction Software Engineering, vol. 18, nº 5, May, in (FENTON, 1994).

SIMONELLI, M. R., 1996, *On fuzzy interactive knowledge*, FSS 80, 159-165.

ZADEH, L. A., 1977, *A theory of approximate reasoning*, Memorandum no. UCB/ERLM 77/58.

ZIMMERMANN, H. J., 1991, *Fuzzy Set Theory and Its Applications*, Kluwer Boston, 2nd revised edition.

ZWICK, R. et al., 1987, *Measures of similarity among fuzzy concepts: A comparative analysis*, Internat. J. Approximate Reasoning 1, in (HSU, 1996).