

Testes de sistemas de informações geográficas com lógica nebulosa

Cláudio Nobre Rapello¹
Orlando Bernardo Filho¹
Vera Maria Benjamin Werneck²

¹ Universidade do Estado do Rio de Janeiro – UERJ
FEN – Faculdade de Engenharia
Rua São Francisco Xavier, 524 – Pavilhão João Lyra Filho - Bloco D - Sala 5022
CEP 20550-900 - Rio de Janeiro – RJ, Brasil
rapello@gmail.com, orlando@eng.uerj.br

² Universidade do Estado do Rio de Janeiro - UERJ
IME - Instituto de Matemática e Estatística
Rua São Francisco Xavier, 524 – Pavilhão João Lyra Filho - Bloco D - Sala 6022
CEP 20550-900 - Rio de Janeiro – RJ, Brasil
vera@ime.uerj.br

Abstract. Testing is a very important activity that is crucial in the development of information systems because most of the software costs could be avoided if better testing was performed. This paper describes a tool for a Testing System named SIT (System of Testing) to support the testing performance of Geographic Information Systems (GIS) through the suggestion of test cases to be used. Geographic Information Systems have the characteristics of spatial referenced persistent data that can generate a great and complex number of test cases. The traditional approaches for designing test cases can be divided into black box (functional) testing and white box testing. At first SIT will treat only the functionality testing method focused on the techniques of equivalence partitioning, boundary value analysis, and cause/effect graphics. SIT also proposes the use of fuzzy logics as a tool to suggest the minimal test case to be used in those Geographic Information Systems. This paper is organized to show an overview of the SIT and presents the treatment of inference of fuzzy logics applied in the system. An example of the application of SIT on a Geographic Information System is also presented in this work to illustrate the benefits of the fuzzy logics module of this tool.

Palavras-chave: software testing, geographic information system, GIS, software engineering, teste de software, sistema de informações geográficas, SIG, engenharia de software.

1. Introdução

Teste de software é um elemento crítico na atividade da Engenharia de Software e na Garantia da Qualidade de Software. Dependendo da criticidade do software, um bom planejamento e uma boa execução do plano de testes podem economizar muito dinheiro, não sendo raro uma organização gastar entre 30% e 40% do esforço total do projeto em testes.

Apesar do uso dos melhores métodos de desenvolvimento e das melhores ferramentas de suporte e treinamento de pessoal, os erros permanecem nos produtos e são essencialmente humanos. Assim, as atividades de revisão e teste representam um papel fundamental para a obtenção de um produto de software com qualidade. Podem-se definir diferentes níveis de testes em relação às diversas fases do processo de desenvolvimento de software, focando classes de erros distintas para se prevenir ou detectar erros cometidos ao longo de todo processo de desenvolvimento (Pressman, 2006).

Por outro lado, na atividade de teste, há uma grande quantidade de limitações, sendo os principais problemas: existência de um conjunto grande de dados de teste para executar todos os caminhos possíveis; retorno do programa com o resultado esperado, porém através de processamentos incorretos, não revelando a existência do erro. O erro pode ser revelado com a entrada de outro valor. Esse tipo de erro é chamado de correção coincidente.

Como o domínio de dados de entrada geralmente é muito grande ou infinito, a atividade de teste pode não mostrar com exatidão que o produto de software está correto, inviabilizando

na maioria das vezes o teste de todas as possibilidades (teste exaustivo). Outro aspecto é a limitação de tempo e recursos destinados aos testes. Sendo assim, duas questões cruciais nesse contexto podem ser formuladas:

- Como selecionar os dados para a realização dos testes?
- Como decidir se um produto foi suficientemente testado?

Para solucionar as questões apresentadas, critérios de teste são estabelecidos, que por sua vez estão incluídos nas técnicas distintas de teste. Deve-se, então, definir e adequar esses critérios e técnicas para a fase de teste a ser realizada. Diversas técnicas clássicas de teste podem ser utilizadas, tais como a técnica funcional (caixa-preta), estrutural (caixa-branca) e a técnica com base em erros.

Diante das dificuldades apontadas para a definição de quais valores das entradas de um sistema devem ser testados, muitos pesquisadores da área têm lançado mão de técnicas da Inteligência Computacional para analisar quais dados são mais relevantes para serem incluídos na massa de teste. Algoritmos Genéticos, por exemplo, é uma das técnicas já experimentadas (Bueno, 2008), todavia uma técnica também possível de ser aplicada seria a Lógica Nebulosa, visto que ela pode ser usada para inferir o quanto se deve testar de uma entrada com base em algumas das suas características mais importantes.

A técnica aqui desenvolvida neste trabalho será aplicada na análise da massa de testes dos Sistemas de Informações Geográficas (SIGs), pois esses são mais abrangentes que os demais sistemas de informação, uma vez que possuem as mesmas características desses últimos, além de incluírem dados referenciados espacialmente, aumento então a complexidade do teste de suas funcionalidades o que vem a tornar mais necessária a criação de forma semi-automática dos testes a serem feitos sobre eles.

O sistema de testes (SIT) foi proposto neste trabalho com o objetivo de sugerir um conjunto mínimo de valores a testar para cada entrada de um Sistema de Informações Geográficas (SIG). Esse conjunto mínimo será identificado através da aplicação das técnicas de testes funcionais associadas a um módulo composto por três sistemas de inferência nebulosos. As seguintes técnicas de teste funcionais foram utilizadas neste trabalho: Particionamento de equivalência, Análise do valor limite e Grafo de causa-efeito.

Neste artigo, descrevemos a visão geral do SIT na Seção 2, apresentando sua arquitetura e algumas características das entradas do SIG a ser testado. Na Seção 3, fornecemos o modelo dos sistemas de inferência nebulosos empregados na construção do SIT. Na Seção 4 destacamos um exemplo de aplicação do SIT e, por fim, na seção 5 expomos as nossas conclusões e sugestões para trabalhos futuros.

2. Visão Geral do Sistema de Testes (SIT)

O sistema de testes (SIT) fornece o valor mínimo a testar para cada entrada de um Sistema de Informações Geográficas (SIG). Esse conjunto mínimo foi associado à aplicação das técnicas funcionais de Particionamento de equivalência, Análise do valor limite, Grafo de causa-efeito e por três sistemas de inferência nebulosos. A razão de ser para a obtenção de tais valores por inferência baseia-se na constatação de que testar todos os valores possíveis de uma entrada para um SIG ou para um software qualquer pode ser muito extenso ou até mesmo inviável, uma vez que algumas entradas podem ter valores infinitos o que levaria a uma óbvia explosão combinatorial (Delamaro, 2007).

O Particionamento de equivalência utiliza classes de equivalência para testar domínios de entrada de um programa, de onde são derivados os casos de teste. O objetivo principal dessa técnica é diminuir os casos de teste, selecionando-se apenas um caso de teste de cada classe, partindo-se do princípio que todos os elementos da classe se comportam de maneira

equivalente. Segundo Myers (Pressman, 2006 e Delamaro, 2007), essa equivalência é aproximada, pois é impossível caracterizar exatamente a equivalência.

A Análise do valor limite é uma técnica que complementa o particionamento de equivalência, utilizando-se os casos de teste nos limites de cada classe de equivalência. Segundo Pressman (2006) e Delamaro (2007), esse critério é bastante relevante, pois os erros costumam ocorrer com mais frequência nos limites dos domínios de entrada.

No Grafo de causa-efeito é verificado o efeito combinado de dados de entrada, montando uma tabela de decisão a partir de causas e efeitos identificados e combinados em um grafo. A partir da tabela gerada são derivados os casos de teste e as saídas (Pressman, 2006 e Delamaro, 2007).

Tomando como base um SIG que possui várias entradas $E1, E2, E3, \dots, En$, a saída S do SIG trata-se de uma função dessas entradas, ou seja, $S = f(E1, E2, E3, \dots, En)$. Para testar o SIG, deve-se observar se para certos valores de $E1, E2, E3, \dots, En$, S apresenta também o valor que se espera (ver Figura 1).

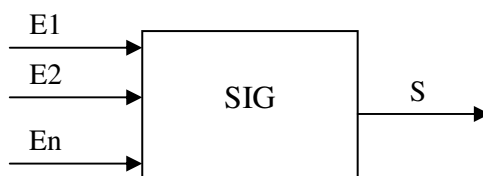


Figura 1. Teste funcional de um SIG.

Inicialmente, pode-se pensar que as entradas possuem dois tipos básicos, a saber:

- (1) Entrada com valores ilimitados em um intervalo ilimitado;
- (2) Entrada com valores ilimitados (ou não) dentro de um intervalo fechado.

Um exemplo de entrada do tipo (1) seria o caso das *strings* que representam nomes de objetos. Para esse exemplo, pode-se observar que todos os valores válidos de *string* são equivalentes, o que leva a concluir que testar um único valor válido já seria suficiente e possível. A preocupação seria apenas pelos valores inválidos de *strings*, como, por exemplo, *strings* nulas, com caracteres especiais, iniciando por números, longas demais etc. Nessa situação, seria necessário testar tais valores anômalos no sentido de observar a reação do sistema. Por outro lado, caso a entrada do sistema fosse dotada de validação, não seria necessário testar os valores anômalos.

No caso de entradas do tipo (2), se o valor da entrada estiver dentro do intervalo fechado, conclui-se que se trata de um valor válido, porém resta saber quantos valores de tal intervalo seria necessário testar.

Nesse ponto, já é possível apresentar a idéia inicial do Sistema de Teste (SIT) a ser desenvolvido, o qual empregará a lógica nebulosa para auxiliar na inferência de valores das entradas de um SIG a ser testado. A Figura 2 apresenta a concepção do SIT.

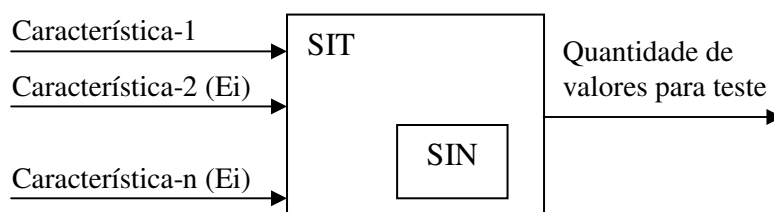


Figura 2. Sistema de Teste (SIT) proposto.

Como se observa na Figura 2, o SIT recebe como entrada as diversas características de uma entrada E_i do SIG sob análise. O SIT internamente será composto de um módulo com três sistemas de inferência nebulosos (SIN) de acordo com a quantidade de características previstas para a entrada E_i do SIG. A figura 3 ilustra as entradas e saídas dos três SIFs que serão utilizados:

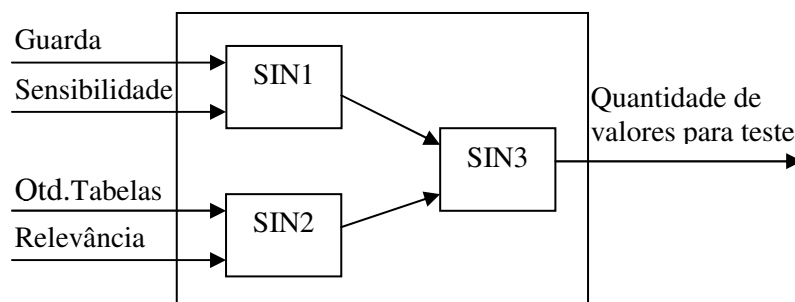


Figura 3. A composição do módulo de inferência nebulosa

O Sistema de Testes (SIT) está sendo desenvolvido na forma de uma dll codificada em Visual Basic 6.0, que contém as classes que manipulam o banco de dados Access 97 e fazem as inferências. Esta dll será instanciada na forma de um objeto ActiveX por páginas ASP (*Active Server Pages*), que oferecerá uma interface *web* bastante amigável com o usuário.

O banco de dados foi dividido em duas partes distintas: na primeira foi modelada toda a parte da lógica nebulosa e, na segunda, toda a parte de cadastro de entradas do SIG.

Na primeira parte, já implementada, estão todas as funções de pertinência e termos lingüísticos de cada antecedente e conseqüente, além das regras da tabela FAM (*Fuzzy Associative Memory*). Na segunda parte, estarão todas as tabelas relativas ao cadastro e manipulação das entradas do SIG.

3. Sistemas de Inferência Nebulosos do SIT

O sistema SIT tem como entrada os seguintes dados: a condição de guarda, o grau de sensibilidade, grau de relevância e quantidade de tabelas acessadas.

Para o caso de entradas com validação, pode-se definir uma relação de guarda G , ou seja, uma entrada E_i , após passar por um processo de validação caracterizado por diversas condições lógicas G , assumiria a forma \hat{E}_i detentora de valores válidos apenas, expurgando quaisquer valores inválidos.

$$E_i \xrightarrow{G} \hat{E}_i \quad (1)$$

A condição de guarda é definida como um valor percentual de 0 a 100 e possui um valor de pertinência entre 0 a 1 para termos nebulosos. Os termos lingüísticos são: fraca (FR), média (M) ou forte (FT). A Figura 4 mostra as funções de pertinência para a condição de guarda.

O grau de sensibilidade denota o quanto a variação de valores de uma entrada causa distúrbios na saída do SIG. Pode haver uma grande variação nos valores e no tempo de processamento da saída caso uma entrada com grau de sensibilidade alto exista no SIG o que vem a indicar uma necessidade de mais testes com essa entrada. Os termos lingüísticos para o grau de sensibilidade são: muito baixo (MB), baixo (B), médio (M), alto (A) ou muito alto (MA). A Figura 5 mostra as funções de pertinência para o grau de sensibilidade.

O grau de relevância define o quanto uma entrada é importante para a saída produzida, levando em conta os principais serviços oferecidos pelo SIG. Podem-se ter entradas muito relevantes ou irrelevantes para a saída esperada. Os termos lingüísticos são: irrelevante (I), opcional (OP), desejável (D) ou obrigatória (OB). A Figura 6 mostra as funções de pertinência para o grau de relevância.

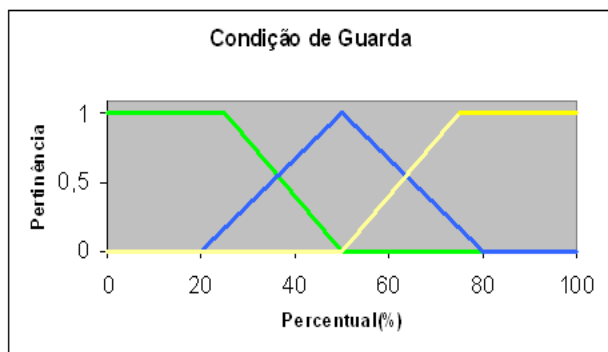


Figura 4. Funções de pertinência para a condição de guarda

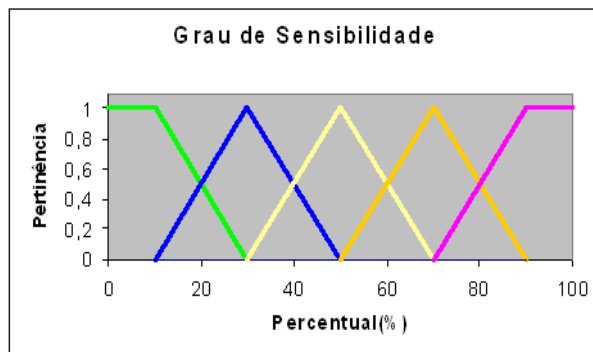


Figura 5. Funções de pertinência para o grau de sensibilidade

A quantidade de tabelas acessadas foi definida pelo intervalo fechado de 0 a 50 tabelas de banco de dados acessadas pela entrada. Os termos lingüísticos são: muito baixa (MB), baixa (B), média (M), alta (A) ou muito alta (MA). A Figura 7 mostra as funções de pertinência para a quantidade de tabelas acessadas.

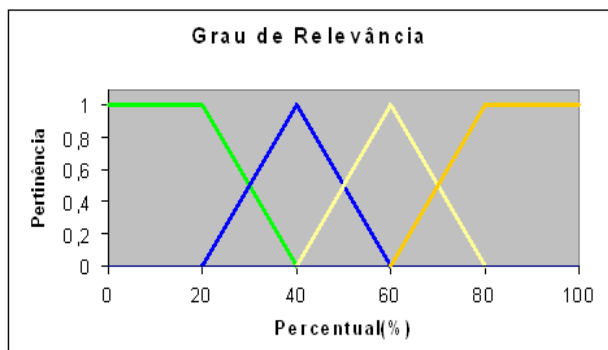


Figura 6. Funções de pertinência para o grau de relevância

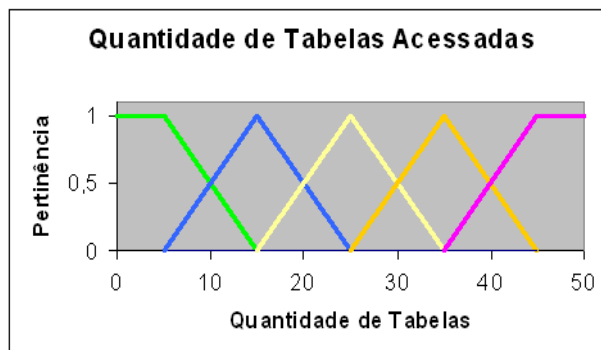


Figura 7. Funções de pertinência para a quantidade de tabelas acessadas

A partir das entradas, foram montadas as tabelas FAM (*Fuzzy Associative Memory*), relacionando os termos lingüísticos. A saída desse primeiro SIN será chamada de **Impacto da Entrada**. Os termos lingüísticos serão: muito pouca (MP), pouca (P), média (M), alta (A) ou muito alta (A). A Tabela 1 mostra os relacionamentos entre guarda e sensibilidade.

A saída do segundo SIN será chamada de **Influência da Entrada**. Os termos lingüísticos serão: muito pouca (MP), pouca (P), média (M), alta (A) ou muito alta (A). A Tabela 2 mostra os relacionamentos entre relevância e quantidade de tabelas.

A saída do terceiro SIN é chamada de **Valores a Testar**. Essa saída é gerada para cada entrada E_i e é dada como um valor entre 0 e 100% em relação ao universo de valores de entrada.

Tabela 1: Guarda X Sensibilidade

		Guarda		
		FR	M	FT
Sensibilidade	MB	P	MP	MP
	B	M	P	MP
	M	M	M	P
	A	A	M	M
	MA	MA	A	M

Tabela 2 - Relevância X Qtd. de Tabelas

		Relevância			
		I	OP	D	OB
QtdTabelas	MB	MP	MP	P	M
	B	MP	P	M	M
	M	P	M	M	A
	A	M	M	A	MA
	MA	M	A	MA	MA

Os gráficos de saída (impacto de entrada, influência da entrada e quantidade de valores para teste) são iguais, podendo ainda sofrer ajustes para um melhor resultado futuro. A Figura 8 apresenta as funções de pertinência dos termos lingüísticos das variáveis impacto de entrada, influência da entrada e quantidade de valores para teste, enquanto a Tabela 3 evidencia as regras nebulosas do SIN3 o qual fornece a saída final da inferência do SIT sobre quantos valores de uma entrada E_i devem ser testados.

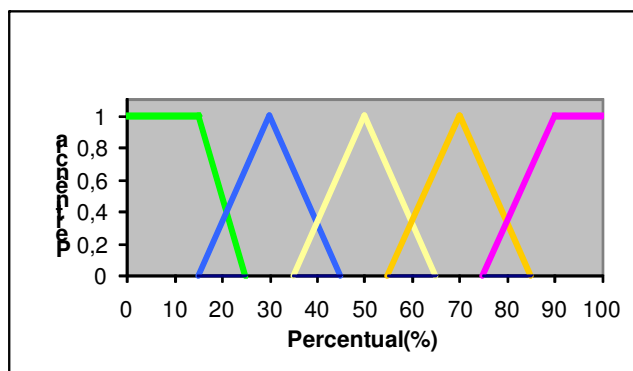


Figura 8. Funções de pertinência dos termos lingüísticos das variáveis Impacto de Entrada, Influência da Entrada e Valores a Testar

Tabela 3: Impacto X Influência

		Impacto				
		MP	P	M	A	MA
Influência	MP	MP	P	P	M	M
	P	P	P	M	M	A
	M	P	M	M	A	A
	A	M	M	A	A	MA
	MA	M	A	A	MA	MA

Para exemplificar, após a execução do SIN3 relativa a uma entrada E_i , sendo composta por um intervalo de valores entre 1 e 40 e, no caso do valor inferido na saída do SIT for 10%, será suficiente testar somente 4 valores dos 40 possíveis. Dois desses quatro valores serão obrigatoriamente os extremos do intervalo considerado para a variação de E_i .

4. Exemplo de Aplicação do SIT

O sistema de teste foi utilizado para avaliar os testes que deveriam ter sido feitos no trabalho de (Salmaso, 2007) no qual tinha sido desenvolvido um sistema de informações geográficas, (chamado de Sistema de Informação Geográfica para Apoio à Tomada de decisão no Setor de Cobrança – SIGATDSC), para consultar as informações dos inadimplentes de um sistema de cobrança. As informações sobre os inadimplentes eram do estado do Rio de Janeiro. O SIG desenvolvido ofereceu consultas sobre o grau de risco da cobrança em um

determinado bairro após adotar uma estratégia de cobrança, avaliar a distribuição geográfica dos inadimplentes por estado e cidades e o índice de pagamento. O SIG empregou um sistema de inferência nebuloso para auxiliar na gestão do acordo dentro do processo de cobrança. Tal sistema fornece como saída o grau de risco da aplicação de uma estratégia de cobrança, levando em conta o resultado do contato com o inadimplente (variável lingüística Acertividade) e o efetivo recebimento do débito (variável lingüística Efetividade).

Essas duas variáveis lingüísticas do trabalho de (Salmaso, 2007) tinham como universo de discurso uma faixa de valores em porcentagem de 0% a 100%. Para testar o seu sistema de inferência nebuloso desenvolvido, seria necessário entrar com valores na faixa de 0% a 100% para ambas as variáveis e observar se o grau de risco calculado (valor também dentro da faixa de 0% a 100%) estaria de acordo com as regras de inferência nebulosas implementadas. A Figura 9 mostra a tabela FAM para as regras do sistema feito por (Salmaso, 2007).

		Efetividade				
GRAU DE RISCO		MB	B	M	A	MA
Acertividade	MB	MA	MA	A	A	M
	B	MA	A	A	M	M
	M	A	M	M	M	B
	A	M	M	B	B	MB
	MA	M	M	B	MB	MB

Legenda: MA-Muito Alto, A – Alto, M- Médio, B – Baixo e MB – Muito Baixo

Figura 9. Tabela FAM do SIGATDSC

As duas variáveis lingüísticas do SIGATDSC possuem as mesmas características, pois são referentes ao mesmo universo de discurso, além de serem fornecidas como entrada da mesma forma no sistema. Sendo assim, será considerado aqui nessa análise para avaliar o comportamento do sistema de teste, a entrada EX do SIGATDSC caracterizada pelo universo de discurso das variáveis lingüísticas Acertividade e Efetividade o qual consiste em uma faixa de valores entre 0% e 100%.

Será necessário agora definir as características dessa entrada EX para que seja possível observar a saída do Sistema de Inferência Nebuloso do SIT, o qual informará quantos valores de EX deverão ser testados. Começando com a característica **Guarda**, os valores de EX são calculados a partir de dados cadastrados diretamente no banco de dados do SIGATDSC pelo usuário com o uso de uma ferramenta de acesso a SGBD externa ao SIGATDSC, sendo assim, não há qualquer proteção sobre os valores inseridos, o que nos leva a concluir que a sua guarda é bem fraca com um valor de 0%. Ainda no diz respeito ao impacto de EX, a sua **Sensibilidade** poderia ficar em torno de 50%, pois EX é o universo de discurso da variável Acertividade ou Efetividade (tanto faz uma vez que a análise é semelhante a ambos os casos), sendo assim, EX influencia a metade do processamento do sistema nebuloso do SIGATDSC.

Para a característica **Relevância**, pode-se estimar um valor de 30%, pois o sistema nebuloso do SIGATDSC corresponde a um terço das suas funcionalidades, enquanto que a **Quantidade de Tabelas Acessadas** é igual a 4, já que as tabelas existentes no SIGATDSC são: TB_USUARIO, TB_INADIMPLENTE, TB_CONTRATO, TB_EMPRESA, TB_TITULO, TB_SITUACAO, TB_ESTADO, TB_ACORDO, TB_PAGAMENTO, TB_TIPOPAG, TB_ESPECIE, TB_POSTO, TB_CIDADE, TB_BAIRRO, TB_TPACIONAMENTO, ANTECEDENTE, REGRA, SEGMENTO, TERMO e

VARIAVEL_LINGUISTICA, sendo que, para a análise do tipo Acertividade x Efetividade do SIGATDSC, apenas as tabelas TB_INADIMPLENTE, TB_EMPRESA, TB_CIDADE e TB_BAIRRO são empregadas na etapa do cálculo de um dos seus índices (Acertividade ou Efetividade).

Entrando com esses valores das características de EX no Sistema de Inferência Nebuloso do SIT, o valor retornado por ele é de 29,9985% de EX, ou seja, dentro da faixa total do EX (0 a 100%), seria necessário testar cerca de 30 valores distribuídos de forma homogênea. Em outras palavras, dever-se-ia testar o valor 0, depois o valor 100/30, o valor $2 \times 100/30$ e assim por diante até o valor 100.

5. Conclusões

Como se observa no exemplo de aplicação do SIT, o seu módulo de inferência nebulosa recomenda que sejam testados cerca de 30 valores para a entrada referente aos índices de Acertividade e Efetividade do SIGATDSC. A princípio, poder-se-ia imaginar que tais valores são muitos, mas se lembrarmos a importância da análise considerada para o SIGATDSC e que os valores da entrada EX afeta fortemente o comportamento do seu sistema de inferência para calcular o grau de risco sobre a possibilidade ou não de manter uma estratégia de cobrança adotada, podemos, dessa forma, concluir que o SIT não exagerou em sugerir 30 valores a serem testados, pois somente assim ter-se-ia a certeza de que o SIGATDSC está fazendo deduções corretas para as suas estratégias de cobrança.

O exemplo de aplicação do SIT apresentado neste artigo concentrou-se no seu módulo de inferência nebulosa, pois é nesse módulo que se encontra implementada a principal contribuição do SIT para resolver o problema da explosão combinatória das massas de teste sugeridas por outras técnicas, todavia, o SIT ainda possuirá outras funcionalidades relativas às entradas que possuem valores ilimitados em um intervalo ilimitado.

Uma vez que o SIT é dotado de “inteligência” para sugerir a massa de testes, mas esses ainda podem ser grandes para os desenvolvedores fazerem de forma manual, seria interessante que o SIT, no futuro, pudesse executar os testes de forma automática.

6. Referências

Aguiar, H.; Oliveira, JR. **Lógica Difusa Aspectos práticos e aplicações**, 1999.

Bueno, P. M. S; Wong, W. E.; Jino, M. Automatic test data generation using particle systems. In: Proceedings of the 2008 ACM symposium on applied computing, 809-814 New York, NY, USA, 2008.

Cox, E. **The fuzzy systems handbook: a practitioner's guide to building, using, and maintaining fuzzy systems**, 1994.

Crespo, A. **Aplicação da Norma IEEE 829 como Mecanismo de Gerência do Processo de Teste de Produtos de Software**, CenPRA, Campinas, 2003.

Davis, C.; Câmara, G. **Arquitetura de Sistemas de Informação Geográfica. Introdução à Ciência da Geoinformação**, cap.3, São José dos Campos, 2001, Disponível em: <<http://www.dpi.inpe.br/gilberto/livro/introd/>>. Acesso em: 30.out.2008.

Delamaro, M.E.; Maldonado, J. C.; Jino, M. **Introdução ao teste de software**, Ed. Elsevier, Rio de Janeiro, 2007.

Pressman, R. S. **Engenharia de Software**, 6ª ed., Rio de Janeiro, McGraw-Hill Interamericana, 2006.

Salmaso, F. V.; Bernardo Filho, O.; Ribeiro, J. A. Sistema de Informação Geográfica para Apoio à Tomada de Decisão no Setor de Cobrança – SIGATDSC, **Anais do XXIII Congresso Brasileiro de Cartografia**, 2007.