

## **APLICAÇÃO DE TÉCNICAS DE VALIDAÇÃO DE *SOFTWARE* EM LABORATÓRIOS DE CALIBRAÇÃO**

A. Silva Ribeiro<sup>1</sup> e J. Alves e Sousa<sup>2</sup>

<sup>1</sup>*Laboratório Nacional de Engenharia Civil, Lisboa, Portugal*

<sup>2</sup>*Laboratório Regional de Engenharia Civil da Madeira, Funchal, Portugal*

### **RESUMO**

A integração de *software* na actividade de laboratórios de calibração, acompanhando o que acontece noutros domínios, tem tido um acréscimo de importância, promovido por exigências de natureza científica e tecnológica, de gestão e de organização.

Essa integração pode, em diversos casos, determinar a qualidade da actividade desenvolvida, não sendo surpreendente que, a normalização de laboratórios acreditados no âmbito de Sistemas de Qualidade, reconheça a sua influência e a realce, introduzindo requisitos específicos, cujo cumprimento estabelece a garantia de que essa influência é adequadamente considerada e quantificada.

Neste enquadramento, reforçou-se o interesse na análise do desempenho e das consequências da integração do *software* nos processos laboratoriais de medição – designadamente, na calibração e nos ensaios – traduzindo-se no incremento do estudo e no desenvolvimento de metodologias que permitam a avaliação pretendida.

Esta análise reflecte-se, por um lado, na estruturação conceptual de métodos e procedimentos de validação e, por outro lado, na sua transposição prática para os casos concretos, envolvendo etapas como a necessária análise e quantificação do risco potencial, a selecção de técnicas e parâmetros métricos e a obtenção de resultados suportados em critérios de avaliação.

Da experiência acumulada em diversos domínios científicos e tecnológicos, são conhecidas inúmeras técnicas de validação, das quais, algumas têm tido uma utilização preponderante como instrumentos de análise de *software* aplicado em laboratórios de calibração. Dessa forma, tem sido possível concretizar a pretendida validação, respondendo às exigências no que se refere ao grau de confiança necessário e, simultaneamente, quantificar – quando relevante – a sua influência, no contexto da própria medição, nos resultados obtidos.

Assim, no presente documento, referem-se sumariamente as etapas de um processo de validação e apresentam-se alguns aspectos associados à selecção, aplicação e análise de resultados que advêm da utilização de algumas das técnicas correntemente utilizadas em laboratórios de calibração.

## **APLICAÇÃO DE TÉCNICAS DE VALIDAÇÃO DE SOFTWARE EM LABORATÓRIOS DE CALIBRAÇÃO**

A. Silva Ribeiro<sup>1</sup> e J. Alves e Sousa<sup>2</sup>

<sup>1</sup>*Laboratório Nacional de Engenharia Civil, Lisboa, Portugal*

<sup>2</sup>*Laboratório Regional de Engenharia Civil da Madeira, Funchal, Portugal*

### **1. INTRODUÇÃO**

O desenvolvimento tecnológico e científico da Metrologia tem conduzido, tal como em outras áreas, a uma crescente intervenção da componente informática no processo de medição de grandezas. Este facto, traduz-se na influência dos resultados obtidos, introduzindo a necessidade/exigência de garantir a “qualidade” dessa intervenção.

No contexto da actividade dos laboratórios de calibração, essa exigência origina a adopção de práticas de validação de software enquadradas como uma das facetas envolvidas na actividade associada aos processos de medição. A sua concretização envolve um procedimento constituído por um conjunto de etapas distintas, das quais uma tem uma elevada importância para o seu sucesso: a selecção e aplicação de técnicas de validação.

Com efeito, desta etapa resulta a conclusão da análise efectuada ao software, traduzida na aceitação ou rejeição do mesmo face ao cumprimento dos requisitos pretendidos. Torna-se evidente que a incorrecta selecção de técnicas pode conduzir a uma situação de risco em que a aceitação é concretizada segundo parâmetros que não são os efectivamente requeridos e, em alternativa, a sua rejeição com base em critérios e parâmetros desadequados representa um custo desnecessário para a entidade que os promove.

Assim, no presente artigo refere-se o enquadramento da etapa de selecção e aplicação de técnicas de validação de software num procedimento genérico adoptado em laboratórios metrológicos e apresentam-se algumas das técnicas correntemente adoptadas no processo de validação e resultados obtidos.

### **2. O PROCEDIMENTO DE VALIDAÇÃO**

O procedimento de validação envolve, em geral, cinco fases [1]: a definição dos requisitos e especificações funcionais do sistema; a análise de risco/integridade e sua classificação numa escala de graduação; a identificação de parâmetros e técnicas de validação; a realização de testes e obtenção de resultados; e a elaboração de um relatório final de conclusão.

As diferentes etapas possuem uma importância repartida, notando-se que, em particular, a segunda – a análise de risco – é preponderante do ponto de vista económico, uma vez que procura estabelecer a medida mais apropriada para a extensão de validação que se irá efectuar e que a terceira etapa – a selecção de técnicas e parâmetros – envolve requisitos de natureza técnica e de conhecimentos específicos para a concretização do processo.

É importante notar que, de acordo com os requisitos da norma NP EN 17025 [2], destinado a entidades acreditadas, a exigência de validação – *confirmar através de evidência objectiva de que foram satisfeitos os requisitos para uma utilização ou aplicação específica* [3] – aplica-se apenas a software comercial modificado e a software desenvolvido internamente (*custom*) entendendo-se que esta orientação se aplica a qualquer outro tipo de laboratórios.

### 3. A SELECÇÃO DE TÉCNICAS

O sucesso da selecção de técnicas de validação depende, fundamentalmente, da correcta avaliação das duas etapas precedentes no procedimento de validação: a identificação das especificações funcionais do software e da análise de risco traduzida na classificação num determinado *grau de integridade requerido ao software* (GIS).

Refira-se que, no contexto da utilização de *software*, se entende que o **risco** [4] corresponde à combinação de eventos anormais ou anomalias e às consequências destes para a falha na operação de um sistema por um utilizador, um operador ou para o ambiente envolvido. A tarefa de quantificar essas consequências nem sempre é simples, necessitando, ela própria de um conhecimento das especificações e características do software e da sua aplicação.

A transposição da análise de risco para uma escala de graduação de GIS, tem como base a definição de 3 parâmetros de avaliação (criticidade; complexidade de processamento; e complexidade de controlo) [1] cada qual com quatro níveis (1 – mínimo a 4 – máximo) e é efectuado um balanço dos resultados obtidos, do qual resulta a determinação do GIS a atribuir ao software (máximo dos níveis parciais). Por exemplo, um software integrado num sistema cuja função é a simples identificação da presença de uma amostra pode ser classificado como GIS 1 enquanto que um software integrado num sistema cuja falha pode envolver risco para o operador é classificado como GIS 4.

Esta etapa é preponderante para a selecção de técnicas a utilizar, dado que existe uma grande diversidade de técnicas disponíveis, com diferentes graus de exigência – e, consequentemente, com diferente custo económico – pelo que se revela útil a existência de tabelas que configurem uma diferenciação de técnicas a utilizar consoante os níveis de GIS. Na Tabela 1 encontra-se desenvolvida uma orientação das técnicas a partir dos níveis de integridade.

Este suporte permite a realização de uma pré-selecção de técnicas, de entre as quais, se deverá adoptar, numa segunda fase, aquelas que são mais adequadas a cada caso particular. Essa segunda fase depende, essencialmente, das especificações e das características individuais de cada *software*. Para concretizar esta tarefa, é importante diferenciar o *software* utilizado em laboratórios de calibração – excluindo sistemas integrados do tipo LIMS (Laboratory Information Management Systems) – de acordo com quatro classes genéricas de classificação:

- *software* embebido em instrumentos de medição ou em sistemas complexos de medição (máquinas de medição de coordenadas, sistemas multicanal de medição de temperatura, ...);
- *software* para processamento de sinais (filtragem, monitorização, análise de dados, ...);
- *software* de bases de dados (aquisição, arquivo e manutenção de informação, ...);
- *software* para cálculo numérico de calibração (determinação de parâmetros estatísticos e incertezas de medição).

Em cada uma destas classes é possível agrupar *software* cujas características e requisitos possuam traços de semelhança e, por essa razão, identificar algumas técnicas de validação de uso preferencial.

**Tabela 1: Técnicas recomendadas para diferentes graus de integridade requeridos (GIS)[1]**

<b>Grau de integridade</b>	<b>Técnicas recomendadas</b>	
<b>1</b>	<ul style="list-style-type: none"> <li>▪ Análise de bases de dados</li> <li>▪ Desempenho temporal <sup>2</sup></li> </ul>	<ul style="list-style-type: none"> <li>▪ Análise de consistência <sup>1</sup></li> <li>▪ Revisão</li> </ul>
<b>2</b>	<ul style="list-style-type: none"> <li>▪ (Técnicas do nível 1)</li> <li>▪ Análise de algoritmos</li> <li>▪ Avaliação de componentes <sup>4</sup></li> <li>▪ Especificações matemáticas <sup>5</sup></li> </ul>	<ul style="list-style-type: none"> <li>▪ Análise de interfaces <sup>3</sup></li> <li>▪ Avaliação estrutural</li> <li>▪ Inspeção</li> </ul>
<b>3</b>	<ul style="list-style-type: none"> <li>▪ (Técnicas do nível 2)</li> <li>▪ Análise em árvore de eventos</li> <li>▪ Avaliação comp. Situações limite</li> <li>▪ Avaliação partições equivalentes</li> <li>▪ Avaliação funcional</li> <li>▪ Estabilidade numérica <sup>10</sup></li> <li>▪ Redes de Petri <sup>11</sup></li> <li>▪ Tabelas de decisão</li> </ul>	<ul style="list-style-type: none"> <li>▪ Análise e avaliação regressiva <sup>6</sup></li> <li>▪ Avaliação em situações de fronteira</li> <li>▪ Avaliação do desempenho <sup>7</sup></li> <li>▪ Conjunto de testes de referência <sup>8</sup></li> <li>▪ Prova de exactidão <sup>9</sup></li> <li>▪ Simulação</li> <li>▪ Tabelas de verdade</li> </ul>
<b>4</b>	<ul style="list-style-type: none"> <li>▪ (Técnicas do nível 3)</li> <li>▪ Análise estática <sup>13</sup></li> <li>▪ Avaliação das declarações</li> <li>▪ Avaliação “back-to-back” <sup>14</sup></li> <li>▪ Revisão do código</li> </ul>	<ul style="list-style-type: none"> <li>▪ Análise temporal crítica <sup>12</sup></li> <li>▪ Comparação com uma estrutura de validação tipo</li> <li>▪ Especificações formais <sup>15</sup></li> </ul>

<sup>1</sup> Comparação com *software* equivalente;

<sup>2</sup> Cumprimento de requisitos e especificações de natureza temporal;

<sup>3</sup> Análise estática de robustez e consistência de módulos e sub-programas;

<sup>4</sup> Análise de módulos para acreditação de software (BS 7923);

<sup>5</sup> Especificações que determinam resultados em função de valores de entrada;

<sup>6</sup> Desenvolvimento e execução de testes que, previsivelmente, induzem erros;

<sup>7</sup> Análise de componentes com influência nos dados de medição/ensaio;

<sup>8</sup> Aplicação a software do tipo *black-boxes*, incorporando o conjunto de dados e de resultados de referência;

<sup>9</sup> Avaliação formal da exactidão com modelos de natureza matemática ou teórica;

<sup>10</sup> Comparação entre a exactidão computacional e a exactidão pretendida, usando variações de pequena amplitude;

<sup>11</sup> Diagramas de transição de estados, incorporando eventos, condições e resultados;

<sup>12</sup> Testes de aspectos críticos de natureza temporal como, por exemplo, de sincronização;

<sup>13</sup> Previsão da execução a partir, por exemplo, da semântica da linguagem utilizada;

<sup>14</sup> Comparação dos resultados usando sistemas computacionais equivalentes sujeitos à mesmas condições de entrada;

<sup>15</sup> Comparação de desempenho com *software* de referência.

No processo de caracterização do software, uma ferramenta auxiliar de utilidade é a constituição de uma lista de questões de base, aplicáveis genericamente, cujo conteúdo permite identificar aspectos particulares do software relevantes para as opções de selecção das técnicas. Uma lista possível inclui as seguintes questões:

- (a) Quais são as especificações / requisitos do utilizador?
- (b) Quais as exigências de validação?
- (c) O Sistema é configurável e qual a sua influência nos resultados obtidos?
- (d) O sistema – e sua estrutura – é acessível ou é fechado (“caixa-negra”) tendo acesso apenas à configuração e aos resultados na saída?
- (e) Quais as características da parametrização admitida?
- (f) Qual o tipo de processamento efectuado?
- (g) Existem e são conhecidos os modelos funcionais e/ou matemáticos para expressar o processamento efectuado?
- (h) Existem mecanismos alternativos para promover a validação – *software* independente validado, existência de outros padrões?
- (i) Existem parâmetros e/ou resultados de referência?
- (j) Que critérios podem ser utilizados para avaliar os resultados da validação?

Finalmente, com base na avaliação do risco potencial traduzido na determinação do GIS, na classificação do software de acordo com as classes apresentadas e com o conhecimento das respostas à lista de questões, seleccionam-se, então, as técnicas para a concretização da validação pretendida.

## 4. ALGUNS EXEMPLOS

A dificuldade natural da tarefa de selecção reside, fundamentalmente, na necessidade de auto-orientação, o que, face a possibilidades múltiplas, pode suscitar dúvidas. Não existindo critérios que determinem, de forma inequívoca, a forma de selecção, afigura-se de particular interesse apresentar, com base na experiência acumulada na actividade de laboratórios metrológicos, algumas das técnicas utilizadas com maior frequência em software que se enquadra nas classes apresentadas anteriormente.

No primeiro caso, relativo a *software embebido em instrumentos de medição ou em sistemas complexos de medição*, a sua caracterização revela que um número significativo de software deste tipo se caracteriza por se enquadrar no que se convencionou designar por “caixa-negra”, conhecendo-se a parametrização, os resultados e, por vezes, as especificações matemáticas das funções envolvidas.

Neste contexto, as técnicas que frequentemente são adoptadas combinam aquela que mais se adapta à natureza deste tipo de *software* – o *conjunto de testes de referência* – com o *teste das especificações matemáticas*, desde que estas sejam conhecidas.

O *conjunto de testes de referência* [1], é uma “técnica baseada no desenvolvimento de conjuntos de dados de referência e de resultados, com os quais é efectuada uma comparação objectiva com valores equivalentes de teste, recorrendo a parâmetros métricos. Esta comparação visa a avaliação do grau de correcção do algoritmo, tendo em vista a exactidão e a adequação aos requisitos associados à sua utilização”, e o *Teste das especificações matemáticas* define-se como uma “técnica baseada em especificações de natureza matemática que determinam os resultados obtidos como função dos valores de entrada. É necessário ter em consideração que, nalguns casos, a forma de implementação computacional poderá ser diferente do tratamento matemático caracterizado na especificação”. Na Figura 1 é ilustrada a combinação das duas técnicas.

Um exemplo comum deste tipo de software encontra-se em instrumentação que utiliza termómetros de resistência de platina e permite efectuar a parametrização de coeficientes da curva de conversão de resistência em temperatura.

Na figura 2 apresenta-se uma interface de um software que permite introduzir parâmetros de curvas que efectuem a referida conversão  $T = f(R)$ .

Este *software* realiza a conversão de valores de resistência eléctrica  $R(t_{90})$  em valores de temperatura ( $t_{90}/^{\circ}\text{C}$ ), de acordo com um algoritmo (1) descrito na ITS-90 (*International Temperature Scale of 1990*) [5], aplicável na gama de  $0^{\circ}\text{C}$  a  $231,928^{\circ}\text{C}$  (ponto de congelação do estanho).

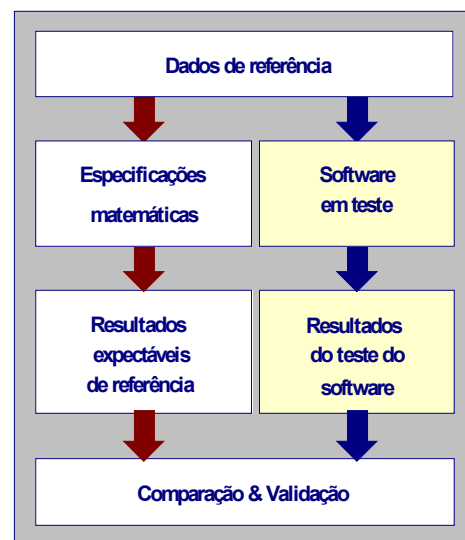


Fig. 1: Avaliação combinada de testes de referência com avaliação pelas especificações matemáticas

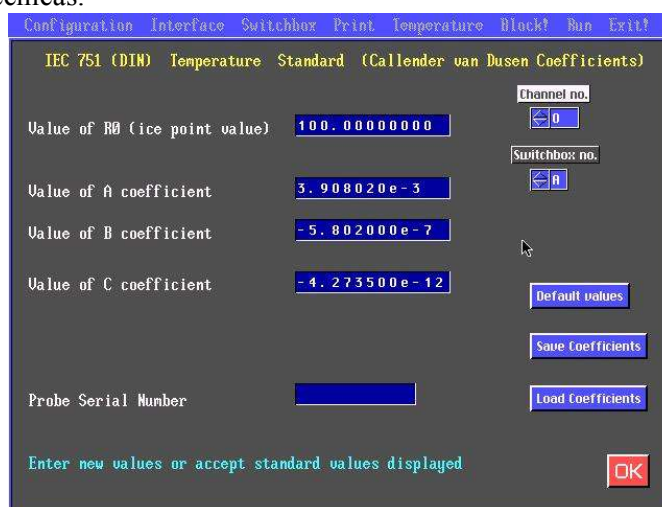


Fig. 2: Exemplo de uma interface (software ASL) para introdução de parâmetros de uma curva de calibração

$$t_{90} / ^\circ\text{C} = D_0 + \sum_{i=1}^9 D_i \left[ \frac{W_r(t_{90}) - 2.64}{1.64} \right]^i \quad (1)$$

As constantes  $D_i$  encontram-se na ITS-90 e os valores de *ratio* da ponte de medida,  $W_r(t_{90})$ , são obtidos usando a equação (2).

$$W(t_{90}) - W_r(t_{90}) = a(W(t_{90}) - 1) + b(W(t_{90}) - 1)^2 \quad (2)$$

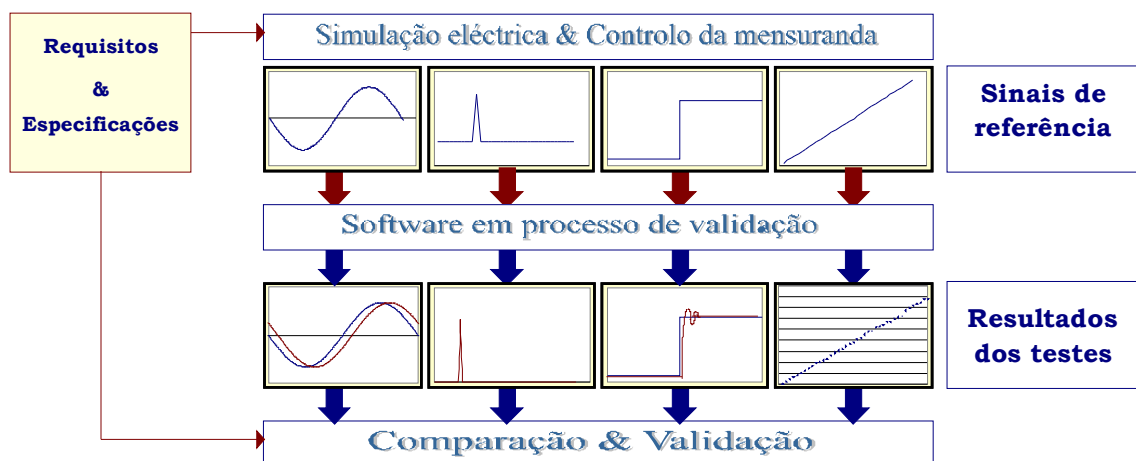
onde o *ratio*  $W(t_{90})$  é dado por  $R(T_{90})/R(273.16 \text{ K})$  e as constantes  $a$ ,  $b$  e  $R(273.16 \text{ K})$ , são determinadas através da calibração do sistema.

Um estudo desenvolvido para dois termómetros de resistência de platina [6] (em canais distintos duma ponte de medida) permitiu obter os valores da Tabela 1, onde a temperatura de referência foi obtida usando os algoritmos (1 e 2) e os valores de calibração ( $a$ ,  $b$  e  $R(273.16 \text{ K})$ ), sendo efectuada uma comparação com as leituras do equipamento e determinado o erro computacional.

**Tabela 1:** Resultados de uma comparação entre valores de referência e valores processados por um software (leituras) relativos a dois termómetros de resistência de platina [6]

Termómetro id.: 362.15 / Canal: 00				Termómetro id.: 363.15 / Canal: 01			
Resistência ( $\Omega$ )	Leituras ( $^\circ\text{C}$ )	Temperatura de referência ( $^\circ\text{C}$ )	Erro computac. ( $^\circ\text{C}$ )	Resistência ( $\Omega$ )	Leituras ( $^\circ\text{C}$ )	Temperatura de referência ( $^\circ\text{C}$ )	Erro computac. ( $^\circ\text{C}$ )
101,000	2,40	2,411	0,011	101,320	3,46	3,466	0,006
111,310	28,98	28,991	0,011	111,349	29,38	29,369	- 0,011
121,991	56,74	56,732	- 0,012	122,085	57,29	57,290	0,000
134,126	88,51	88,509	- 0,001	134,150	88,91	88,909	- 0,001
144,943	117,07	117,071	0,001	144,829	117,11	117,112	0,002
155,833	146,06	146,056	- 0,004	155,748	146,17	146,165	- 0,005
167,020	176,08	176,078	- 0,002	166,651	175,40	175,397	- 0,003

No segundo caso, relativo a *software para processamento de sinais*, as aplicações mais comuns envolvem a utilização de aspectos associados ao desempenho temporal do sistema, tendo como base as características dinâmicas da mensuranda. Neste contexto, são frequentes aplicações que envolvem as situações de análise de frequências, de monitorização do comportamento de grandezas, de comunicação de informação proveniente de instrumentação e de sincronização, entre outras.



**Fig. 3:** Validação de Software envolvendo processamento de sinais

Neste tipo de situações, é comum o recurso à técnica de *avaliação do desempenho temporal*, “técnica baseada na avaliação do cumprimento de especificações e requisitos de natureza temporal definidas para a ferramenta computacional”. A validação resulta, ainda, do recurso suplementar a técnicas de avaliação de situações de risco potencial – associadas, por exemplo, a valores limite e de

fronteira – como a *análise de interfaces*, “técnica de análise dinâmica, que recorre a conjuntos de dados de teste adequados à avaliação do desempenho das variáveis em teste quer para valores normais quer para valores extremos, visando a determinação de inconsistências (em atributos de variáveis globais, incorrecção de parametrizações, arquivo incorrecto de variáveis estáticas e dinâmicas, entre outras)”.

Um exemplo de aplicação consiste na monitorização de grandezas como a temperatura e a humidade relativa, efectuadas em diferentes pontos do espaço, por exemplo, para obter perfis térmicos dos mesmos. Nesse tipo de situações é necessário garantir, por um lado, uma adequada transmissão de dados e, simultaneamente, a sincronização –dentro de limites adequados ao problema em causa – da informação proveniente dos diferentes sensores.

O terceiro caso referido, de *software de desenvolvimento de sistemas relacionais de bases de dados*, envolve, no processo de validação, uma diversidade maior de técnicas devido aos distintos recursos que, em geral, se podem utilizar. Essas técnicas são agrupadas numa técnica de base designada por *análise de bases de dados*, cuja descrição refere: “técnica baseada na avaliação da estrutura da base de dados e dos métodos de acesso, de modo a garantir que a gestão de dados e variáveis é consistente e mantém um elevado nível de integridade. Esta avaliação é particularmente importante em módulos que envolvem o arquivo e a gestão de quantidades significativas de dados, sendo necessário assegurar a sua protecção relativamente a situações de escrita sobreposta e introdução de valores incorrectos”.

Em geral, a concretização das tarefas de validação enquadram-se em 3 vertentes globais da aplicação: a estrutura relacional da base de dados; a interface de operação; e a gestão da informação. Dentro de cada uma destas, encontram-se componentes específicas de avaliação, as quais, são apresentadas na tabela seguinte, as mais significativas.

**Tabela 2:** *Vertentes e Componentes envolvidas num processo de validação de Sistemas Relacionais de Bases de Dados*

Estrutura relacional	Conteúdo de bases de dados Relacionamento entre bases de dados Campos de indexação Coerência interna de informação (dimensões de campos, memória disponível e requerida, parametrização de campos, etc.)
Interface	Segurança e protecção Análise de situações limite e de fronteira Selecção de pesquisa de dados Apresentação de dados (consulta, impressão, transferência, ...) Actualização de dados Operacionalidade de comandos e botões Actividades de Ajuda ao utilizador
Gestão de informação	Interdependência entre campos de informação Execução de funções e macros Execução de operações de natureza especial (numérica, data, ...) Concepção de listas de selecção e avaliação da coerência do conteúdo das listas Arquivo de dados (segurança, protecção) Modificação, transferência e limpeza de informação Avaliação de características: manutenção, portabilidade, eficiência e outras.

Um exemplo de aplicação consiste nos sistemas de bases de dados relacionais aplicados à gestão metrológica de instrumentação, as quais, têm como principais funções, o arquivo de informação relativo à instrumentação, designadamente, a informação relacionada com o historial de calibrações e o apoio à utilização da mesma – por exemplo, na constituição de curvas de correcção – e aos planos de calibração e de manutenção. Na figura seguinte, encontra-se uma imagem de uma interface dum sistema deste tipo, contendo informação acerca de um determinado instrumento de medição.

MENU: Consulta & Actualização por número informático

Pesquisa por N° informático: 226.01  Actualizar registo Sair

---

Equipamento: Craveira digital

Marca: MITUTOYO Modelo: CD0-015D

---

N° Informático: 226.01 N° de Série: 0003890 N° inventário:

---

Unidade orgânica: CIC/NOM/LCAM Contacto: João A. Fernandes

Sala: 0180B  3204 

---

Situação: Operacional Tipo de calibração: Calibração no LCAM/LNEC

Próxima calibração: 31/08/05 Prazo de calibração: 12 meses

---

Consulta Outros Dados Grandezas associadas Calibração/Historial 

---

INFO: Índice primário: Equipamento; Índices secundários: Unidade orgânica e N° informático

Record: 383 of 1592

Fig. 4: Exemplo de uma interface num Sistema Relacional de Bases de Dados

Finalmente, no que se refere ao software desenvolvido para executar cálculo numérico relacionado com medições e calibração, o tipo de exigências envolvido encontra-se associado a problemas de natureza numérica, sendo conhecidos os modelos matemáticos que os suportam. Consoante a natureza dos modelos em causa, existem diferentes exigências de validação, sendo as mais frequentes relacionadas com a *estabilidade numérica* – por ex., quando se dispõe de modelos com risco potencial de “overflow” ou de “divisão por zero”, entre outras – com as situações de fronteira – relacionadas com limitações de parâmetros e de resultados, por exemplo – e com a complexidade da transposição dos modelos para rotinas de programação ou para funções – por exemplo, com diferenciação ou integração de funções, etc.

Nos diferentes casos, aplica-se uma estratégia equivalente à apresentada para o primeiro caso – software embebido em instrumentação – combinando a técnica de testes de referência com as *especificações matemáticas* (Fig. 1), com a definição cuidada dos dados de referência.

Neste contexto, não se pode dissociar a constituição deste tipo de análise com a vertente particular do problema que consiste na correcta selecção de dados de referência para os testes. Como breve referência a este aspecto, é importante referir que o processo específico de constituição de dados de teste pode, ele próprio, ser objecto de um estudo específico como o apresentado em [7].

Com efeito, a necessidade de consistência entre os dados gerados ou definidos e as especificações funcionais de cada problema é crucial para uma correcta avaliação do software em causa. Na referência citada, apontam-se 3 elementos fundamentais de comparação entre os resultados do teste e os resultados de referência, constituindo a base das *métricas de qualidade*: as diferenças absolutas entre os resultados de teste e de referência; a exactidão computacional suportada pelo *hardware*; e o “grau de dificuldade” do conjunto de dados de referência.

Destes elementos referidos, resulta a adopção de 3 métricas para a avaliação:

- i. Uma *métrica absoluta do afastamento*, expressa pela diferença entre os valores dos resultados obtidos e os resultados de referência;
- ii. Uma *métrica relativa do afastamento*, expressando o número de algarismos coincidentes;
- iii. Uma *métrica relativa ao desempenho*, constituída pelos factores que determinam as duas anteriores.

Na referência [7] encontra-se um exemplo particular de uma *métrica de desempenho*, a qual tem a seguinte expressão:



$$p(\mathbf{b}) = \log_{10} \left( 1 + \frac{d(\mathbf{b})}{k\eta} \right)$$

Nesta,  $d(\mathbf{b}) = \text{RMS}(\mathbf{d}) = \|\mathbf{d}\|_2 / \sqrt{n}$ , sendo  $n$  o número de elementos do vector,  $\mathbf{d} = \mathbf{b}^{(\text{test})} - \mathbf{b}^{(\text{ref})}$ , onde  $\mathbf{b}^{(\text{test})}$  é o vector de resultados do teste e  $\mathbf{b}^{(\text{ref})}$  é o vector de resultados de referência,  $\eta$  é o menor número positivo representável tal que o valor  $(1+\eta)$ , calculado usando a aritmética computacional, excede a unidade e  $k$  designa-se por “grau de dificuldade do problema”, sendo um factor definido pelo analista, que depende da natureza do problema e da dimensão associada à escala em que se contabilizam as grandezas.

Esta métrica de desempenho, tem como resultado a indicação do número de algoritmos significativos que se perdem quando comparado com um algoritmo estável segundo critérios de optimalidade. Os seus resultados aproximam-se de 0 quando o desempenho do software em teste se aproxima do esperado (algoritmo estável) e toma valores próximos de  $k$  quando se verifica a perda de  $k$  algoritmos significativos.

## 5. CONCLUSÕES

O processo de validação de software, pela intervenção crescente que tem nos processos de medição, não pode ser negligenciado. Os custos económicos e o risco envolvido justificam que a extensão da análise seja suportada em metodologias criteriosas que efectuem um adequado balanceamento dessas duas vertentes.

As tarefas envolvidas neste processo nem sempre se afiguram de fácil execução, com particular destaque para situações em que não existe um conhecimento completo do software sujeito à avaliação e quando o processamento de informação tem um elevado grau de complexidade. Nesses casos, o sucesso desta avaliação é fortemente dependente do conhecimento das especificações funcionais dos problemas envolvidos, da capacidade técnica para gerar dados de referência, determinar resultados de referência e sujeitar o sistema ao processamento da informação pretendida.

Como resultado final, existirá um incremento substancial na confiança relativamente aos resultados obtidos, bem como, a possibilidade de, através do conhecimento de deficiências ou limitações do software testado, aumentar o grau de conhecimento sobre a intervenção deste na medição e das suas reais capacidades para a aplicação em causa ou para outras aplicações pretendidas.

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] RIBEIRO, A. - *Validação de Ferramentas Computacionais. Elementos Sobre a Sua Incidência em Metrologia*. Informação Científica e Técnica. LNEC, Lisboa, 2001.
- [2] NP EN ISO/IEC 17025: *Requisitos gerais de competência para laboratórios de ensaio e calibração*, Instituto Português da Qualidade, 2000.
- [3] NP EN ISO 9000: *Sistemas de gestão da qualidade. Fundamentos e vocabulário*, Instituto Português da Qualidade, 2000.
- [4] WICHMANN, B.A. - *Software in Scientific Instruments, Measurement Good Practice Guide N° 5*. NPL, Teddington, U.K, 1999.
- [5] *The International Temperature Scale of 1990, ITS-90*, NPL, UK, 1991.
- [6] COX, M. G. AND HARRIS, P. M. - *The design and use of reference data sets for testing scientific software*. NPL Report, Teddington, U.K, 1998.
- [7] RIBEIRO, A., CASTRO, M. AND CORREIA, J. - *Development and Validation of Computational Tools for Temperature Testing*. 8<sup>th</sup> International Symposium on Temperature and Thermal Measurement in Industry and Science, Berlim, Alemanha, 2001.