

Desenvolvimento de um simulador web para incertezas em medição por coordenadas

Autores

Alvaro Jose Abackerli
Nivaldi Calonego Junior
Luis Augusto Consularo

Apoio Financeiro

Fap

1. Introdução

Atualmente, usuários de CMM precisam declarar as incertezas de medição segundo o ISO GUM (ISO, 1995). Como a determinação da incerteza em máquinas de medir não é necessariamente uma tarefa simples, diferentes Máquinas Virtuais de Medir por Coordenadas (*Virtual Coordinate Measuring Machine – VCMM*) têm sido desenvolvidas para esta finalidade (METROSAGE, 2006 e SCHWENKE, 2006).

A Figura 1 traz o Diagrama do Processo de Medição ilustrando esquematicamente tanto a medição real quanto a sua simulação. Os conectores cheios da figura representam o fluxo das informações na configuração da medição real. Os conectores tracejados representam o procedimento análogo da simulação.

Conforme ilustra a figura, a configuração completa da medição inclui a definição de um sistema apalpador, da máquina, do ambiente, da peça e das suas informações de manufatura. Na medição real verifica-se ainda a influência de fatores externos como o operador, a fixação da peça, além dos efeitos das decisões do operador sobre a estratégia geral de medição. Estes fatores externos influenciam os resultados mas não são necessariamente controlados para compensação e correção durante uma medição real.

Em síntese, tanto na medição real quanto na simulação existem fatores que interferem nos resultados e que não são completamente considerados, seja por limitações técnicas da sua caracterização experimental, limitações teóricas ligadas à sua modelagem ou mesmo limitações econômicas associadas à sua monitoração e controle.

Testes recentes em VCMM comerciais mostraram a importância da compreensão do usuário sobre a medição real e sobre processo de simulação, mostrando também que a fidelidade da simulação em relação ao mundo real é tão importante quanto a exatidão dos resultados numéricos gerados por estes softwares simuladores.

Por isso, atenção deve ser dada à inclusão de operações guiadas no projeto de interfaces de simuladores, auxiliando o usuário na simulação que visa o cálculo das incertezas.

2. Objetivos

Neste trabalho são discutidos alguns aspectos do projeto da interface de um simulador de máquina de medir, no qual foi aplicado o conceito MVC – Model/View/Controller com o objetivo de desacoplar as partes integrantes do sistema para buscar melhores níveis de usabilidade do software desenvolvido e maior fidelidade da simulação.

3. Desenvolvimento

Pela complexidade do problema simulado pela VCMM o projeto da sua interface é elaborado focalizando dois aspectos particulares do desempenho, a fidelidade e a usabilidade do simulador.

Para discutir a fidelidade da simulação e a usabilidade da VCMM primeiro é necessário definir estes conceitos no contexto da presente discussão. Fidelidade da simulação é aqui definida como “a habilidade do software de simulação em representar fielmente na simulação os detalhes de uma medição real.” (PHILLIPS et al, 2003). Usabilidade é aqui definida como a medida da facilidade de aprender e usar o sistema do ponto de vista do usuário. Além disso, envolve a eficácia das funcionalidades do sistema no suporte às tarefas do usuário e sua capacidade de restaurar a operação normal mediante erro do usuário (SWEBOK, 2004).

A discussão anterior revela a complexidade do problema de simulação que é aqui caracterizado para fins de projeto pelos seus três requisitos principais:

- Número de parâmetros de configuração: além do grande número, eles podem variar no processo de simulação dependendo das opções selecionadas de simulação para simulação. O usuário deve sempre estar atento em relação à informação fornecida e da proximidade entre os cenários de medida simulado e real;
- Ordem de configuração: refere-se à seqüência da configuração e influências nos outros passos de configuração conforme Figura 1;
- Conhecimento do usuário: está relacionado com seus conhecimentos prévios sobre o problema de medição real e sua curva de aprendizado do processo de simulação. Embora isso não seja muito diferente de uma situação para outra, aqui o problema se torna um pouco mais complexo porque em geral o usuário não tem outros meios externos para verificar o resultado gerado pelo simulador.

4. Resultados

A evolução das técnicas de desenvolvimento de aplicações induziu a concepção de sistemas do tipo MVC - Model/View/Controller. A técnica MVC tem por finalidade minimizar o acoplamento existente entre a interface e o processamento dos dados numa aplicação, podendo ser aplicado no desenvolvimento de interfaces complexas assumindo que: “Model” corresponda ao subsistema de representação dos dados da interface; “View” corresponda a forma que o usuário visualiza (ou percebe) os dados; e “Controller” é a lógica de acesso ao “Model” a partir de um “View”. Um diagrama de classes para a configuração da VCMM em notação UML, segundo o modelo MVC é ilustrado na Figura 2.

As classes *VCMMSetup*, *VCMMMainForm*, *UserInterfaceState* correspondem ao “Model”, “View” e “Controller”, respectivamente. As setas indicam o sentido e o tipo da relação entre os objetos das respectivas classes. As linhas cheias com setas abertas indicam que ocorre uma associação, isto é, um objeto instanciado de *VCMMMainForm* tem acesso aos objetos instanciados de *VCMMSetup* e *UserInterfaceState*. As setas abertas com linhas pontilhadas indicam uma relação de dependência entre as classes. A classe

XMLDocument cria uma visão de dados para o simulador baseada em um “XML Schema”, que padroniza a comunicação com o núcleo de simulação. Os itens a seguir detalham o modelo UML da Figura 2 para a elaboração do “pattern” para as interfaces.

4.1 O “Model”

As classes que compõem o “Model” encapsulam o acesso aos dados, objetivando minimizar o acoplamento e viabilizar o re-uso do modelo em futuras extensões. Ele determina a granularidade dos objetos, definindo partições hierárquicas no conjunto de dados para a configuração da VCMM. A elaboração desse modelo considerou 283 variáveis de simulação para a composição da classe *VCMMSetup*, abrangendo partições que compreendem a configuração de todos blocos ilustrados na Figura 1.

Os estados dos objetos instanciados de *VCMMSetup* que compõem uma configuração dependem dos valores digitados nos respectivos formulários das interfaces. Esse modelo é tratado de duas maneiras distintas em relação à visão dos dados (“View”). Uma visão é a do usuário que configura os dados das simulações e a outra é visão do simulador. O item a seguir detalha as visões consideradas para o modelo de interfaces VCMM.

4.2 O “View”

A primeira visão dos dados corresponde à forma como eles são inseridos pelo usuário, permitindo caracterizar o “View” como o conjunto de janelas de navegação que dão acesso aos campos dos formulários. Nesse modelo foi utilizado o “pattern decorator” (Gamma, 1999) como técnica de modelagem. Antes de definir a configuração dos dados para a simulação, o usuário pode optar por operações de carga, impressão, inspeção de itens configurados, tipo de simulação, help, dentre outros, que devem estar disponíveis em todos os formulários, mas que possuem funcionalidades diferentes nos respectivos formulários.

Todas as classes da camada que define o “View” são abstratas porque são usadas em formulários cujas funcionalidades que dependem do tipo de usuário. Esses diferentes comportamentos das interfaces (formulários) são definidos em função da habilidade do usuário na configuração de VCMMs, sendo estes classificados em quatro grupos básicos conforme discutido abaixo.

Do ponto de vista do simulador, há uma visão que trata da transposição dos dados da interface para um documento XML válido. A validação desses dados é feita em relação às restrições impostas por máquinas reais de medir usando uma linguagem regular codificada no “XML Schema” já mencionado.

4.3 O “Controler”

A concepção do módulo de controle do comportamento da interface do programa ou “Controler” se baseia na especificação de uma Máquinas de Estados Finitos – FSM. Considerando que a VCMM possui 283 variáveis para a sua configuração e que a ordem desses dados pode não ser alternada, tem-se uma FSM com a arranjo de 283 possíveis transições. Esse tipo de dificuldade estimulou Harel (1987) no desenvolvimento de uma extensão da FSM, denominada *Statechart* que além de facilitar a especificação de sistemas de tempo real possibilita a agregação de subconjuntos de estados de uma FSM em superestados.

No problema em estudo, o tipo de usuário classificado nos grupos *Newbie*, *Trainee*, *Intermediary*, *Expert* determina um conjunto de ações diferentes associadas aos eventos da interface. Essas categorias estão associadas ao tipo de controle que se pretende ter sobre as ações do usuário durante a configuração da VCMM.

A categoria *Newbie* abrange usuários sem experiências que pretendem se familiarizar com as interfaces de configuração. Neste caso, são apresentadas estratégias de medições diferentes que implementam seqüências válidas de configuração mas não é permitida a criação de novas configurações, o usuário deve apenas

analisar cada uma das configurações e ver os seus resultados na simulação. Usuários *Trainee* são aqueles que têm permissão para a modificar valores das configurações, mas são tratados na estratégia de medição. Usuários classificados como *Intermediary* podem modificar estratégias de medição e editar parâmetros de configuração. Usuários Expert não têm restrições.

Em síntese, as categorias *Newbie*, *Trainee*, *Intermediary* e *Expert* determinam quais eventos o usuário pode gerar, significando que o conjunto de ações é o mesmo para todos os usuários mas o conjunto de transições da FMS é específico por categoria de usuário.

5. Considerações Finais

Os aspectos de usabilidade almejados no projeto aqui discutido dependem fortemente das camadas definidas pelo “View” e pelo “Controler”. O projeto do “View” mostrou que a partição das variáveis de configuração em diferentes formulários tem potencial para manter o usuário atento às diferentes etapas da configuração.

A estrutura hierarquizada dos formulários permitiu a criação dos diferentes grupos de usuários. Para cada grupo de usuários foram definidos os diferentes controles sobre as suas ações, dependendo das respectivas experiência em máquinas de medir e no processo de simulação.

No modelo proposto, a fidelidade está principalmente relacionada com o “Model”, que armazena todas as variáveis necessárias para executar a simulação. O modelo mostra como criar as partições dos dados de acordo com os passos da configuração ilustrados na Figura 1, permitindo ainda sua extensão para incorporar tantas variáveis adicionais quantas forem necessárias para aprimorar a fidelidade da simulação, conforme anteriormente definido.

Finalmente, as interfaces gráficas criadas podem ser livremente modificadas para melhorar diversos aspectos da comunicação com o usuário, sem que com isto sejam afetadas as suas funcionalidades. Além disso, para trocar as ações associadas aos elementos da interface basta alterar os métodos implementados nas classes derivadas, o que aumenta a flexibilidade para evoluir o aplicativo mediante demandas do processo de simulação.

Referências Bibliográficas

- SWEBOK Guide to the Software Engineering Body of Knowledge (SWEBOK): trial version. Abran,A., Moore,J.W., Bourque,P., Dupuis,R. & Tripp,L.L. (eds.), (IEEE Computer Society, Los Alamitos - CA, 2004). http://www.swebok.org/ironman/pdf/SWEBOK_Guide_2004.pdf Access on 24/08/2006.
- Gamma, E; et al. *Design Patterns: Elements of reusable object-oriented software*, ed. 17, Addison-Wesley, 1999. ISBN 0-201-63361-2
- HAREL, David. *Statechart: a visual formalism for complex systems*. Science of Computer Programming, 8, North-Holland : Elsevier Science Publisher ,1987. pp. 231-274.
- ISO Guide to the Expression of uncertainty in measurement. 2^a ed. International Organization for Standardization, Genève, Switzerland: ISO, 1995.
- METROSAGE (EUA). Metrosage Plc (Org.). **PUNDIT/CMM**. Disponível em: . Acesso em: 23 ago. 2006.

PHILLIPS, S.D. et al. The validation of CMM task specific measurement uncertainty software. 2003. ASPE. ASPE Proceedings of the Summer Tropical Meeting on Coordinate Measuring Machines.

SCHWENKE, H. (Germany). Physikalisch-technische Bundesanstalt (ptb) (Org.). Virtual CMM. Disponível em: . Acesso em: 23 ago. 2006.

Anexos



