
Um Modelo de Apoio ao Projeto de Design de Interfaces de Ambientes de Aprendizado

Raquel Oliveira Prates, Rosa Maria Videira de Figueiredo, Catharine Ferreira Bach

Instituto de Matemática e Estatística, UERJ, 6o. andar , R. São Francisco Xavier, 524
Maracanã, Rio de Janeiro, RJ, Brasil, 20550 – 013

{raquel,rosa}@ime.uerj.br

***Abstract.** The development of good software interfaces is a challenge to developers. Even more so, when the interface also has to allow for learning. In this paper we present GOMSQ, a task-model adapted to an educational context. This model is a first step towards providing interface designers and educators with models, methods and techniques that can support them in the development of interfaces for educational softwares.*

***Resumo.** O desenvolvimento de interfaces de qualidade é um desafio. No caso de ambientes educacionais este desafio é ainda maior, uma vez que ela deve fomentar o aprendizado de um conteúdo pelo usuário. Neste trabalho apresentamos o GOMSQ, um modelo de tarefas adaptado para um contexto educacional. Este modelo é um primeiro passo na direção de se oferecer a projetistas e educadores envolvidos no desenvolvimento de ambientes educacionais modelos, métodos e técnicas que apoiem o desenvolvimento de interfaces neste contexto específico.*

1. Introdução

Um dos grandes desafios da educação na atualidade é integrar a computação às práticas de ensino exercidas em escolas e universidades. Um dos esforços nesta direção tem sido o desenvolvimento e uso de ambientes de aprendizado. Ambientes de aprendizado são aqueles que dão apoio ao trabalho feito para se alterar o conhecimento do aluno sobre um assunto [Carey et al. 1998]. O design de softwares educacionais traz novos requisitos para o processo de desenvolvimento e avaliação de software e, conseqüentemente, novos desafios para a indústria de software.

Um dos aspectos fundamentais ao sucesso de um software é a qualidade da sua interface com os usuários. A interface envolve todos os aspectos de um sistema com o qual o usuário mantém contato [Moran, 1981] e logo tem um impacto direto no uso que as pessoas podem ou conseguem fazer dele. O bom design de uma interface depende de diversos fatores que incluem desde a satisfação subjetiva do usuário até o retorno do investimento. A pesquisa na área de Interfaces Humano-Computador (IHC) tem por objetivo fornecer explicações e previsões para fenômenos de interação usuário-sistema e resultados práticos para o design da interface de usuário [ACM SIGCHI, 1992].

A área de IHC tem sido caracterizada, principalmente, pelas abordagens de base cognitiva [Norman e Draper, 1986; Preece et al. 2002]. Estas abordagens têm raízes

comuns com as áreas de psicologia cognitiva, ciência cognitiva e inteligência artificial que estudam a cognição, isto é, o processo pelo qual se pode adquirir conhecimento, e aplicam suas teorias na compreensão das capacidades e limitações da mente dos usuários. Dentro destas abordagens a principal métrica da qualidade de uma interface é a sua usabilidade. Uma interface tem boa usabilidade quando os usuários têm facilidade em aprendê-la e utilizá-la, conseguem se lembrar como atingir os objetivos que têm e, fazem isso com alto grau de produtividade, satisfação pessoal e eficiência.

Enquanto as abordagens cognitivas enfatizam o processo de compreensão do usuário, as abordagens semióticas enfatizam a expressão do designer, e desta forma podem ser vistas como abordagens complementares. A Semiótica é uma disciplina que estuda processos de significação e comunicação [Eco, 1976]. Assim, as abordagens semióticas de IHC concebem uma aplicação computacional como um ato de comunicação que inclui o designer no papel de emissor de uma mensagem para os usuários dos sistemas por ele criados [Nadin 1988; Andersen et al. 1993; de Souza 1993; Oliveira e Baranauskas, 1998; de Souza et al., 2001].

Na Engenharia Semiótica [de Souza 93; de Souza et al. 2001], em particular, a interface de um sistema é explicitamente tratada como uma mensagem enviada pelo designer aos usuário. Esta mensagem tem por objetivo comunicar ao usuário quem o designer acredita que ele (o usuário) seja, o que ele pode fazer com o sistema e como interagir com o sistema para fazê-lo. O usuário deve ser capaz de perceber a resposta a estas perguntas através da sua interação com o sistema. Como a mensagem enviada pelo designer (i.e. a interface) também é capaz de se comunicar com os usuários através da troca de mensagens, ela é considerada um artefato de meta-comunicação. Além disso, ela é unilateral e única, uma vez que o usuário recebe a mensagem concluída e não pode dar continuidade ao processo de comunicação naquele mesmo contexto de interação. Nesta abordagem a principal métrica da qualidade de uma interface é sua comunicabilidade. Uma interface tem boa comunicabilidade quando consegue transmitir com sucesso ao usuário a mensagem do designer, de forma que o usuário compreenda para que o sistema serve, quais as vantagens de usá-lo, como funciona, e quais os princípios gerais de interação embutidos na interface.

Com base na fundamentação teórica, tanto cognitiva, quanto semiótica, foram desenvolvidos modelos, técnicas e métodos para apoiar as etapas de projeto e avaliação de interfaces. No entanto, os modelos, técnicas e métodos desenvolvidos para interfaces de software genéricos, nem sempre são apropriados para interfaces de sistemas de aprendizado, uma vez que estes não apenas acrescentam novos requisitos ao projeto a ser desenvolvido, mas muitas vezes alteram requisitos existentes. Neste artigo propomos o GOMSQ que tem por objetivo possibilitar ao educador a representação de alguns aspectos do apoio ao aprendizado durante a modelagem da tarefa de aprendizado. Para atingirmos este objetivo estendemos um conhecido método de modelagem de tarefas, o GOMS [Card et al., 1983, Lee, 1993], utilizando para isto o método de avaliação de comunicabilidade. Como avaliação preliminar do GOMSQ utilizamo-no para o desenvolvimento de um sistema de apoio ao aprendizado de Programação Linear, o AMPLIAR.

Na próxima seção apresentamos alguns dos desafios do desenvolvimento de interfaces educacionais e apresentamos também uma adaptação da abordagem da

Engenharia Semiótica para o contexto de softwares educacionais que busca dar uma descrição mais precisa deste domínio. Na seção seguinte apresentamos uma breve descrição do modelo GOMS e do método de avaliação de comunicabilidade, e então apresentamos o GOMSQ. Em seguida, descrevemos o projeto AMPLIAR e os indicadores obtidos através dele. Finalmente, discutimos as conclusões do trabalho e os próximos passos a serem seguidos para dar continuidade à pesquisa.

2. Desafios de Interfaces Educacionais

A qualidade de uma interface educacional requer que ela preencha os requisitos de usabilidade e comunicabilidade como a interface de qualquer software, entretanto, isto não é suficiente para atestar a sua qualidade. Para ser considerada de boa qualidade, a interface educacional deve conseguir atingir com sucesso o seu principal objetivo, proporcionar o aprendizado de um conteúdo por um usuário. Isto traz implicações não apenas para a etapa de avaliação de interfaces, mas também para o design destas interfaces. Uma importante consequência desta constatação é o fato de que as técnicas de ensino e objetivos de aprendizado devem estar representados na interface de forma a guiar as atividades dos alunos. Faz-se necessário então que o educador participe ativamente do projeto da interface.

Na abordagem da Engenharia Semiótica a responsabilidade do educador é definir o modelo educacional no qual a atividade de aprendizado vai se basear, e a sua participação ativa no projeto da interface o torna co-autor da mensagem (i.e. a interface) a ser transmitida ao usuário [Martins et al. 2000]. Neste caso a mensagem a ser transmitida é a resposta às perguntas: (a) Qual o conteúdo que eu pretendo ensinar a você?; (b) Que tarefas você deve executar para conseguir aprendê-lo? e (c) Como interajo com o sistema para conseguir executar as tarefas que preciso e aprender o conteúdo oferecido?. O educador definirá as respostas às perguntas (a) e (b), enquanto o projetista definirá a resposta à pergunta (c). A Figura 1 ilustra a abordagem da Engenharia Semiótica para interfaces educacionais.

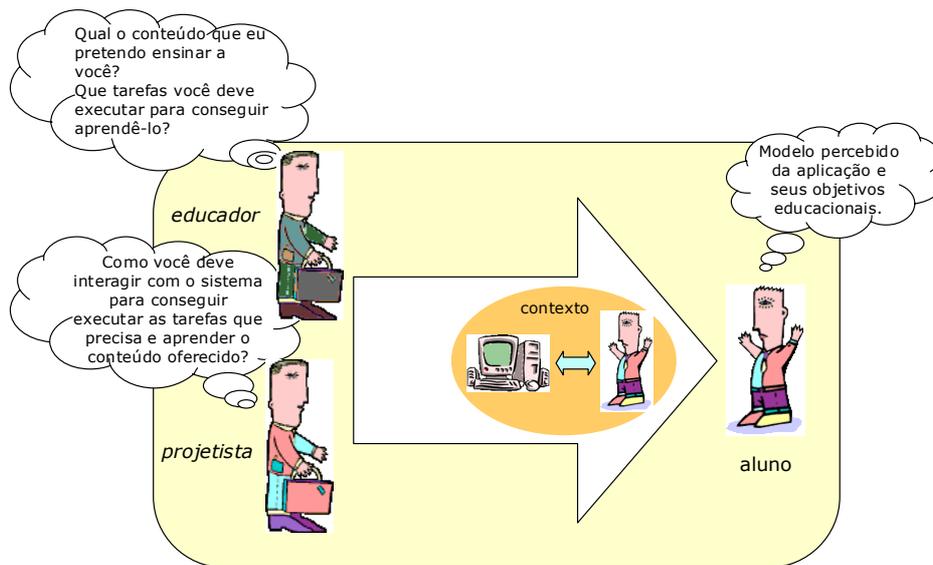


Figura 1 - Abordagem da Engenharia Semiótica para o design de interfaces educacionais.

Como o principal objetivo de uma interface educacional é o aprendizado, muitas vezes ao se definir as melhores respostas para as perguntas (a) e (b) acima, se vai contra as diretrizes para que se forneça uma boa resposta a (c). Por exemplo, uma das principais diretrizes de interfaces é que o software previna erros dos usuários, entretanto, em um contexto educacional pode-se fomentar um erro do aluno para garantir o aprendizado de um determinado ponto [Lewis et al., 1998]. Assim, as interfaces educacionais requerem que modelos, técnicas e diretrizes sejam revistos para considerar suas particularidades. Neste trabalho apresentamos uma proposta que representa um passo nesta direção.

3. Modelo de Tarefa Adaptado para um Contexto Educacional: GOMSQ

O GOMSQ (*Goals, Operators, Methods, Selection Rules and Questions*) tem por objetivo apoiar projetos de interfaces educacionais, oferecendo um modelo para se representar as tarefas a serem executadas em uma atividade de aprendizado e as comunicações sobre estas [Bach 2001, Bach et al. 2001]. Para isso, este modelo estende o GOMS simplificado [Lee, 1993] com base nas expressões de comunicabilidade [Prates et al. 2000]. A seguir apresentamos brevemente o modelo GOMS e o método de avaliação de comunicabilidade.

3.1. Modelo GOMS

A modelagem de tarefas é uma etapa do processo de design de uma interface que tem por objetivo permitir ao designer formalizar as tarefas a serem disponibilizadas para o usuário na interface sendo projetada. Dentre os vários modelos e formalismos propostos para se atingir este objetivo tem-se o GOMS (*Goals, Operators, Methods, and Selection Rules*) [Card et al., 1983]. Este modelo de base cognitiva pretende representar o comportamento dinâmico do usuário durante a sua interação com o computador. Para isto ele representa o comportamento humano através de três subsistemas de interação: o perceptual (auditivo e

visual), o motor (movimentos braço-mão-dedo e cabeça-olho), e o cognitivo (tomadas de decisão e acesso à memória). Este comportamento é definido por:

- metas (goals), aquilo que o usuário deseja fazer, e podem ser decompostas numa hierarquia de sub-metas;
- operadores (operators), que são os atos perceptivos, motores ou cognitivos básicos que os usuários devem executar para afetar o ambiente da tarefa;
- métodos (methods), que são seqüências de passos para se atingir uma meta; e
- regras de seleção (selection rules), expressões do tipo “condição-ação”, que devem ser utilizadas para selecionar um determinado método, sempre que houver mais de um disponível, para atingir uma meta. .

Uma versão simplificada e mais abstrata do GOMS foi proposta [Lee, 1993] com o objetivo de reduzir o grau de conhecimento de psicologia cognitiva necessário para aplicação do método, e torná-lo mais independente das escolhas dos elementos de interface. Estes objetivos são atingidos através da não especificação dos operadores envolvidos nas tarefas. O trabalho aqui apresentado se baseia nesta proposta do GOMS simplificado.

3.2. Método de avaliação de comunicabilidade

O método de avaliação de comunicabilidade simula a comunicação do usuário para o designer sobre os problemas de interação que o usuário teve ao utilizar a interface. Isto é feito através de um pequeno conjunto de expressões que o usuário potencialmente usaria para se exprimir em uma situação onde acontece uma ruptura na sua comunicação com o sistema [Prates et al. 2000]. A aplicação do método pode ser dividida em duas etapas: a coleta de dados e a análise destes dados. Para coletar os dados prepara-se um teste com usuários e grava-se a interação do usuário com a aplicação, isto é um filme da interação, usando para isto um software de captura das ações do usuário, como por exemplo o SnagIt ®. Além do filme da interação pode-se enriquecer os dados com anotações do observador durante o teste, o filme em vídeo do teste e uma entrevista com o usuário.

A análise dos dados é feita com base no filme da interação gerado, utilizando as demais informações para entender melhor o contexto e possíveis linhas de raciocínio dos participantes do teste. Esta análise é composta por três passos:

1. Etiquetação: ver gravações da interação e atribuir a expressão apropriada nos momentos de ruptura da interação. Por exemplo, quando o usuário percorre opções do menu à procura de uma determinada função, então se associaria a esta porção da interação a expressão "Cadê?"; ou se o usuário abre uma janela e a fecha imediatamente, se associaria a expressão "Epa!"; ou ainda se o usuário pára seu cursor sobre um botão da barra de ferramentas a espera de uma dica, se associaria a expressão “O que é isso?”
2. Interpretação: tabular e consolidar a informação obtida, ou seja, as expressões atribuídas no passo de etiquetação, associando-as a ontologias de problemas de interação ou diretrizes de design. Diferentes ontologias poderiam ser usadas pelo

avaliador. As classes de problemas identificados são navegação, atribuição de significado, falha de execução da tarefa, e incompreensão ou recusa de *affordance*¹.

3. Perfil Semiótico: Interpretar a tabela (passo 2) dentro do quadro teórico da Engenharia Semiótica, em uma tentativa de se reconstruir a meta-mensagem sendo transmitida pelo designer ao usuário através da interface.

3.3. GOMSQ

Como vimos, o GOMS permite ao designer representar as tarefas possíveis de serem executadas com o software e as ações necessárias para executá-las. A nossa motivação em utilizá-lo para a modelagem de tarefas de ambientes de aprendizado é permitir ao educador definir as tarefas que o aluno deve executar para conseguir alterar o seu conhecimento sobre aquele conteúdo, ou seja, definir a estratégia de ensino desejada. Além de executar a atividade prevista, durante o aprendizado o aluno pode necessitar, ou querer, se engajar em uma comunicação sobre o conteúdo (e.g. pedir um esclarecimento) ou sobre a própria atividade sendo executada (e.g. discutir o objetivo da atividade). O educador ao projetar um ambiente de aprendizado pode desejar disponibilizar aos alunos mecanismos que apoiem esta comunicação sobre a atividade. No entanto, o GOMS não prevê uma representação que dê conta da comunicação sobre a tarefa sendo descrita.

Para permitir ao educador representar a comunicação do aluno com o sistema² de aprendizado sobre a tarefa sendo executada, nos baseamos nas expressões do método de avaliação de comunicabilidade e estendemos o GOMS simplificado. Ao permitir a representação desta comunicação o GOMSQ (*Goals, Operators, Methods, Selection Rules and Questions*) pretende permitir ao educador evitar rupturas no processo de aprendizado utilizando o sistema, ou seja, permitir que se preveja o apoio a ser oferecido ao aluno nos momentos em que ele interrompe o processo por necessitar de mais informações ou esclarecimentos sobre a atividade ou conteúdo [Bach 2001, Bach et al. 2001].

Para poder utilizar o GOMSQ, o primeiro passo do educador é partir do conjunto de expressões de comunicabilidade e selecionar quais ele gostaria de representar no seu ambiente³. Cabe ressaltar que as expressões originais de comunicabilidade foram definidas para descrever as rupturas de interação e não para um ambiente de aprendizado. Desta forma, ele pode necessitar definir novas expressões que não fazem parte deste conjunto⁴. Feito isso, o educador (se desejar, juntamente com o projetista da interface)

1 Termo que se refere às propriedades percebidas e reais de um artefato, em particular as propriedades fundamentais que determinam como este artefato pode ser utilizado [Norman, 1988].

2 Cabe ressaltar que o sistema age como porta-voz da equipe de design, uma vez que toda comunicação que ele pode fazer foi prevista e definida pela equipe.

3 A nossa expectativa é de que a maior parte das expressões selecionadas serão perguntas, por isso a extensão Questions ao título do método.

4 Já foi feito um trabalho que propõe o uso das expressões de comunicabilidade para o projeto do sistema de ajuda. Neste trabalho, um novo conjunto de expressões foi proposto, contendo novas expressões e excluindo algumas das originais [Silveira 2002]. Talvez o educador possa partir deste conjunto para definir o conjunto de expressões desejadas para apoiar a sua atividade de ensino.

parte para a modelagem da atividade de ensino desejada. Esta modelagem consiste de três passos:

- (1) fazer a modelagem de tarefas do sistema de aprendizado utilizando o GOMS simplificado;
- (2) a cada meta ou sub-meta identificada na atividade de ensino, acrescentar o conjunto de expressões definidas como desejáveis;
- (3) associar a cada expressão, a cada passo, as respostas fornecidas pelo educador para serem apresentadas ao aluno quando ele “expressar” aquela expressão.

Note-se que o GOMSQ não inclui decisões sobre as telas que vão compor o sistema, os elementos de interface a serem utilizados, ou a disposição dos mesmos na tela. Entendemos isto como sendo um benefício, uma vez que isto torna o modelo independente da sua implementação na interface e permite que ele sirva como base para várias opções de ambiente.

Na próxima seção apresentamos o AMPLIAR, o Ambiente de Apoio ao Aprendizado em Programação Linear, e ilustramos a aplicação do método aqui descrito.

4. AMPLIAR

O ensino do Método Simplex para resolução de problemas de Programação Linear, requer que uma grande quantidade de operações algébricas sejam executadas para resolver um problema de grande porte e isto, muitas vezes, pode desmotivar o aluno a aprender o método, ou fazer com que ele priorize as operações algébricas em detrimento ao entendimento do processo. Além disso, essa grande quantidade de operações algébricas pode levar o aluno a cometer erros durante a resolução de um problema. Corrigir estes erros implicaria em refazer os passos da resolução a partir do erro identificado, o que envolveria novas operações algébricas. Esse alto custo para refazer um problema quando identificado um erro leva, muitas vezes, o aluno a desistir da resolução do problema. Soma-se a isso o pouco tempo disponível para treinamento do método em sala de aula e a dificuldade do aluno de realizar tal treinamento em horário extra-classe devido a falta de apoio para identificação de erros cometidos durante a resolução. Desta forma identificou-se a necessidade de se oferecer aos alunos um ambiente que fosse capaz de apoiá-los no processo de aprendizado do Método Simplex e complementar o ensino sendo feito em sala de aula.

Embora estejam disponíveis no mercado diversos pacotes para resolução de problemas de Programação Linear [CPLEX, XPRESS, What's Best e Lindo] eles não atendem de forma plena às necessidades de aprendizado dos alunos. Isto porque o objetivo desses softwares é a resolução eficiente de problemas lineares de grande porte. O AMPLIAR [Bach, 2001; Prates et al., 2002] tem por objetivo o apoio ao aprendizado do Método Simplex de forma interativa e didática. Para isso ele pretende oferecer ao aluno um ambiente no qual ele pode se concentrar no entendimento dos passos a serem executados e nas decisões envolvidas a cada passo, e não nas operações algébricas necessárias. Para permitir o foco do projeto nas atividades de aprendizado e na compreensão destas, a modelagem do AMPLIAR foi feita utilizando o GOMSQ, o que nos permitiu ter uma avaliação preliminar da aplicação deste modelo.

4.1. Projeto do AMPLIAR utilizando o GOMSQ

Conforme descrito na seção 3.3 o primeiro passo para a utilização do método GOMSQ é a definição das expressões a serem utilizadas. O objetivo do educador era que o aluno utilizasse o AMPLIAR no horário extra-classe para consolidar os conceitos ensinados em sala. Desta forma, as tarefas a serem modeladas eram os passos necessários para a execução do Método Simplex, e a comunicação sobre elas visava sanar as possíveis dúvidas do aluno durante a resolução do método, além de permitir uma compreensão mais profunda sobre o método ensinado. Partindo do conjunto original de expressões de comunicabilidade o educador identificou o seguinte subconjunto que representaria as dúvidas esperadas (ou mais freqüentes) dos alunos na execução do método:

- **O que é?:** O aluno quer saber o que é determinado elemento que aparece na resolução do Simplex;
- **E agora?:** O aluno quer saber qual o próximo passo para a resolução do Método Simplex;
- **Como?:** O aluno quer saber como ele deve fazer para executar uma determinada tarefa durante a resolução do Simplex;

Para representar a comunicação que buscava dar um apoio ao entendimento do método (e não somente à sua execução) o educador propôs duas novas expressões⁵:

- **Por quê?:** O aluno quer saber por que determinado elemento aparece, por que determinado elemento possui o valor apresentado ou por que ele deve executar o passo indicado pelo software para prosseguir na execução do Simplex;
- **Eu entendi?:** O aluno quer apresentar o que entendeu para verificar se está entendendo corretamente⁶.

Então, utilizando este conjunto de expressões, o projetista e o educador, juntos, modelaram a tarefa de aprendizado do AMPLIAR. Para isso, eles definiram cada uma das metas e ações necessárias para se resolver o problema aplicando o Método Simplex. Para cada uma destas metas acrescentaram o conjunto de expressões definido acima, e o educador definiu a resposta a ser oferecida pelo sistema ao aluno para cada uma delas. A Figura 2 apresenta parte do modelo gerado utilizando o GOMSQ, mostrando as metas necessárias e a descrição do conjunto de expressões para uma delas.

(1) Colocar o problema no formato padrão;
(2) Definir uma solução básica viável inicial;
(3) Definir qual variável entrará na base;

- **O que é a variável que entrará na base?**
R: É uma variável não pertencente a base atual que ao assumir um valor maior que zero permitirá reduzir o valor da função objetivo.
- **O que é a base?**
R: A base B é uma submatriz da matriz A, com $B \in R^{n \times n}$ e $\det(B) \neq 0$, tal que $B^{-1}b \geq 0$. A solução

⁵ Estas expressões não fazem parte do conjunto original de expressões do método de avaliação de comunicabilidade.

⁶ Esta pergunta foi prevista, mas não foi implementada na primeira versão do AMPLIAR, pois requer uma análise da resposta do usuário.

- associada a tal base é denominada solução básica viável e é um ponto extremo da região viável.
- **Como** faço para escolher a variável que entrará na base?
R: Procure dentre todas as variáveis fora da base uma variável x_k com $z_k - c_k > 0$.
 - **E agora**, o que faço depois de escolher a variável que entrará na base?
R: Escolha a variável que vai sair da base, no lugar da qual a variável escolhida para entrar ficará.
 - **Por que** tenho que escolher a variável que entrará na base?
R: Porque a solução básica associada à base atual não é ótima, e é possível diminuir o valor da função objetivo ao fazer uma variável $x_k > 0$, ou seja, ao fazer x_k entrar na base, se $z_k - c_k > 0$.
- (4) Definir qual variável sairá da base;
(5) Apresentar a solução do problema.

Figura 2 - Parte do modelo de tarefas do AMPLIAR utilizando o GOMSQ

A Figura 3 a seguir mostra uma das telas do AMPLIAR, e o ato do aluno de fazer uma pergunta ao sistema sobre como executar um passo. Observe que no menu principal estão disponíveis todas as perguntas que podem ser feitas pelo aluno naquele momento. Este menu está disponível durante todos os passos do método, e as perguntas disponíveis a cada momento dependem do passo do método que o aluno está executando. Na Figura 4 mostramos a resposta apresentada ao aluno.

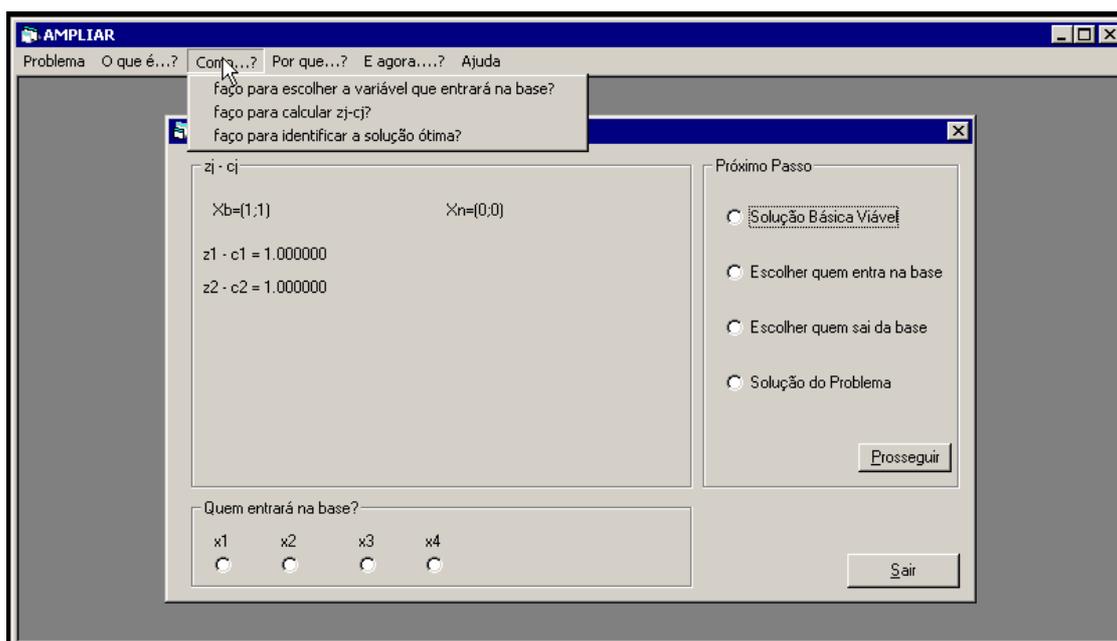


Figura 3 - Tela do AMPLIAR, mostrando perguntas disponíveis aos alunos na etapa de escolha de variável que entrará na base.

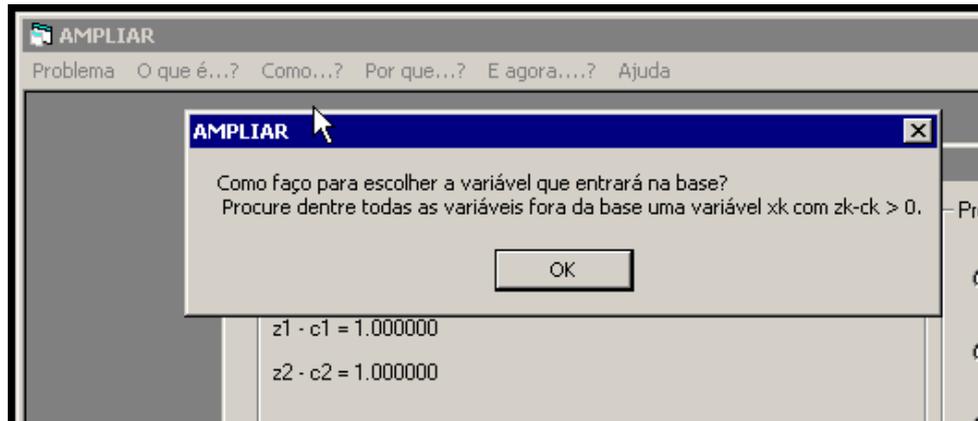


Figura 4 - Tela do AMPLIAR mostrando resposta oferecida ao aluno após uma pergunta.

5. Avaliação preliminar

Uma avaliação preliminar do sistema AMPLIAR foi feita no curso de Programação Linear do Instituto de Matemática e Estatística da UERJ. O sistema foi utilizado no laboratório pelos alunos do curso com o acompanhamento do educador e do projetista. O objetivo da avaliação era tanto ter indicadores sobre o uso que os alunos fariam do AMPLIAR, quanto do seu projeto baseado no GOMSQ, ou seja, das perguntas disponibilizadas para eles. Esta avaliação produziu uma lista de problemas a serem acertados e ajustes a serem feitos no AMPLIAR.

Em termos de indicações, tanto sobre o AMPLIAR quanto o GOMSQ, a principal observação do teste foi relacionada com o uso que os alunos fizeram das perguntas. Em um primeiro momento, apesar de as perguntas estarem disponíveis no menu principal, os alunos não perceberam a possibilidade de as fazer para o próprio sistema. Muitas vezes eles faziam exatamente as perguntas que estavam disponíveis no sistema ao educador, que então mostrava a possibilidade de fazê-las para o sistema. A partir daí os alunos recorriam ao sistema e apenas em caso de dúvidas sobre a resposta do sistema, recorriam ao educador. Esta observação nos leva a crer que o apoio comunicativo projetado utilizando o GOMSQ e implementado no AMPLIAR será útil aos alunos na consolidação do método em horário extra-classe. Esta observação também aponta para a necessidade de a interface enfatizar mais para o aluno a possibilidade de se fazer perguntas ao sistema.

A observação do uso dos alunos e suas dúvidas permitiram a identificação de vários problemas de usabilidade e comunicabilidade tanto da interface, quanto das respostas oferecidas. Esta avaliação preliminar permitiu a definição dos próximos passos no desenvolvimento do sistema AMPLIAR e da pesquisa. Em relação ao AMPLIAR, o educador ajustou as respostas às perguntas com base nas dificuldades apresentadas aos alunos. A interface do sistema foi avaliada por mais um método de avaliação para identificar outros problemas de usabilidade e comunicabilidade e uma nova versão está sendo gerada. Esta nova versão permitirá uma avaliação mais extensa do AMPLIAR e do GOMSQ.

6. Conclusões

O GOMSQ se propõe a dar um apoio mais direcionado às necessidades de representação de projetistas de interfaces de ambientes de aprendizado. O modelo atinge este objetivo na medida em que permite a expressão não apenas das ações a serem executadas para se alcançar o aprendizado, mas também a representação da comunicação necessária, ou desejável, relacionada a esta tarefa, entre o porta-voz do educador (o sistema) e o aluno. Uma primeira avaliação do GOMSQ foi feita utilizando-o no projeto do sistema AMPLIAR. O projeto mostrou que é possível utilizar o GOMSQ para fazer a modelagem da tarefa de aprendizado. Além disso, a avaliação preliminar do sistema AMPLIAR fornece indicadores dos benefícios de se levar o educador a pensar e representar a comunicação sobre esta tarefa.

O sistema AMPLIAR permite apenas uma avaliação inicial do GOMSQ. Para poder se apreciar o real uso que os alunos farão das expressões disponibilizadas durante o aprendizado, é necessário primeiramente tornar o sistema AMPLIAR mais robusto. Feito isto, poder-se-á observar melhor o uso que os alunos farão do sistema, das perguntas oferecidas sobre o conteúdo e avaliar junto a eles o apoio que o AMPLIAR e a comunicação educador-aluno, projetados através do GOMSQ, foram capazes de oferecer.

Um ponto a ser avaliado sobre o GOMSQ é em que contextos educacionais e para que tipos de ambientes ele é apropriado. Para se ter indicadores dos contextos em que ele poderia ser aplicado seria interessante observar a sua contribuição para o ensino de outros métodos utilizando a mesma estratégia do AMPLIAR e também utilizando estratégias de ensino distintas. Seria ainda necessário avaliar aplicações que buscassem o apoio do aprendizado de outros tipos de conteúdos (e.g. conceitos, experimentos científicos, etc).

O GOMSQ é um passo na direção de se apoiar especificamente projetistas de interface e educadores trabalhando no projeto de ambientes educacionais. O modelo gerado para o sistema AMPLIAR deu indícios da sua utilidade, tanto durante o design do software auxiliando a colaboração entre projetista e educador, quanto no apoio oferecido ao aprendizado pelo sistema baseado nele. Os indícios apresentados pelo uso do GOMSQ motivam novas investigações de como adaptar ou definir métodos, modelos e técnicas de IHC para o contexto educacional.

Agradecimento

As autoras agradecem à UERJ o apoio ao Projeto AMPLIAR e aos alunos do curso de Programação Linear sua valiosa participação na avaliação deste.

Referências

- ACM SIGCHI (1992) "Curricula for human-computer interaction". Technical report, ACM, NY, 1992. Disponível on-line em <http://www.acm.org/sigchi/>.
- Andersen, P.B.; Holmqvist, B.; Jensen, F.F. (eds.) (1993) *The Computer as Medium*. Cambridge University Press.
- Carey, T.T., Harrigan, K.A. & Palmer, A. (1998). "Mediated Conversations for Cognitive Apprenticeship". In Bruckman, A.S., Guzdial, M., Koldner, J.L. & Ram, A. (Eds.),

-
- ICLS 1998, Proceedings of the International Conference of the Learning Sciences, Atlanta, GA. (pp. 299-301). Charlottesville, VA.
- Bach, C. F. (2001). AMPLIAR: AMbiente de aprendizado em Programação LineAR. Projeto de Final de Curso, Dept. de Computação, IME, UERJ, Rio de Janeiro, Brasil.
- Bach, C. F., Prates, R. O.; Figueiredo, R. M. V. (2001) “GOMSQ: Um método de modelagem de tarefas para sistemas educacionais baseado em comunicabilidade”. Anais do IV Workshop em Fatores Humanos em Sistemas Computacionais, 290-293.
- Card, S.; Moran, T. & Newell A. (1983) The Psychology of Human-Computer Interaction, Hillsdale, NJ: Lawrence Erlbaum Associates.
- CPLEX: Otimization Software. Disponível na Internet em www.cplex.com. Site consultado em Janeiro/2001.
- de Souza, C. S. (1993), The Semiotic Engineering of User Interface Languages, International Journal of Man-Machine Studies, v. 39, pp. 753-773.
- de Souza, C. S., Barbosa, S. D. J., and Prates, R. O. (2001), A Semiotic Engineering Approach to User Interface Design. Knowledge Based Systems. Amsterdam, v. 14, n. 8, pp. 461-465.
- Eco, U. (1976) A Theory of Semiotics. Bloomington, IN: Indiana University Press.
- Lee, G. (1993) Object-Oriented GUI Application Development. NJ: Prentice Hall.
- Lewis, C., Brand, C., Cherry, G., Rader, C; (1998) Adapting User Interface Design Methods to the Design of Educational Activities. CHI 98, Los Angeles, pp 18-23.
- Martins, I. H., Santos, N., Prates, R. O. “Cooperação entre educador e projetista no design de interfaces para ambientes educacionais”. Cardernos do IME, série Informática, UERJ, Rio de Janeiro, Vol. 9, Dezembro, 2000, pp 3-10.
- Moran, T. (1981) “The Command Language Grammars: a representation for the user interface of interactive computer systems”. International Journal of Man-Machine Studies, 15, 3-50.
- Nadin, M. (1988) “Interface Design and Evaluation – Semiotic Implications”. In Hartson, R. e Hix, D. (eds.), Advances in Human-Computer Interaction, Vol. 2, 45-100.
- Norman, D., Draper, S. (eds.) (1986). User Centered System Design. Hillsdale, NJ. Lawrence Erlbaum.
- Norman, D. (1988) Psychology of Everyday Things. HarperCollins Publishers.
- Oliveira, O. L., Baranauskas, M. C. C. (1998). “A Semiótica e o Design de Software”. Relatório Técnico IC-98-09. <http://www.dcc.unicamp.br>
- Prates, R.O.; de Souza, C.S.; Barbosa, S.D.J.; “A Method for Evaluating the Communicability of User Interfaces.”. Interactions 7, 1. NY: ACM Press, 31-38, 2000.
- Prates, R. O.; Figueiredo, R. M. V., Bach, C. F. (2002) “Projeto de um Ambiente de Aprendizado para Complementar o Ensino do Método Simplex em Sala de Aula”. Em Carvalho, L. M. (ed.), História e Tecnologia no Ensino de Matemática.

Preece, J.; Rogers, Y.; Sharp, E. (2002). Interaction Design: Beyond Human-Computer Interaction. New York, NY: John Wiley & Sons.

Silveira, M. S. (2002). Metacomunicação designer-usuário na interação humano-computador: design e construção do sistema de ajuda. Tese de doutorado, Dept. de Informática, Pontifícia Universidade Católica do Rio de Janeiro, RJ, Brasil.

XPRESS: Modeling and optimization software. Disponível na Internet em www.dash.co.uk. Site consultado em Janeiro/2001.

What's Best e Lindo. Disponível na Internet em www.lindo.com. Site consultado em Janeiro/2001.