

Protótipo Inicial da Interface Gráfica de um Provedor de Teoremas

LETÍCIA CARVALHO PIVETTA FENDT¹
ARTHUR BUCHSBAUM²
ADRIANA FIRMINO DE ANDRADE¹
NATEL BARREIROS¹

¹ULBRA – Universidade Luterana do Brasil
ILES-JP – Instituto Luterano de Ensino superior de Ji-Paraná
Caixa Postal 271 – Ji-Paraná – RO – CEP 78960-000
{leticia,aandrade,natel}@inf.ulbrajp.com.br

²UFSC – Universidade Federal de Santa Catarina
INE – Departamento de Informática e Estatística
Florianópolis – SC – CEP 88040-900
arthur@inf.ufsc.br

Resumo: O presente trabalho descreve a atual fase de desenvolvimento do Sistema Provedor de Teoremas para a Lógica Clássica, na qual está em ênfase o desenvolvimento de uma interface adequada para utilização desse sistema por iniciantes na área de Lógica de Predicados. A interface está sendo desenvolvida em separado do sistema a que se destina, através da metodologia de Protótipos Descartáveis. Este trabalho apresenta a descrição do protótipo inicial e a verificação de conformidade com requisitos de usabilidade, segundo a lista de checagem Ergolist. A verificação dos requisitos possibilitou a identificação de uma série de pontos que devem ser sanados, para posterior integração sistema-interface e submissão do sistema à aceitação dos usuários.

Palavras-chave: Lógica Clássica, Método dos Tableaux, Interface e Ergonomia.

1. Introdução

A automatização do raciocínio, que é um dos ramos da Inteligência Artificial, trata da simulação mecânica do processo de pensar realizado pelo homem. A importância do raciocínio automático está no fato de liberar o homem de processos repetitivos, rotineiros e estafantes, permitindo que ele se dedique a tarefas que necessitem de um certo nível de criatividade, atualmente um tanto quanto distantes da máquina.

Uma forma de automatização do raciocínio é definida através da escolha de uma lógica para representação do conhecimento e a utilização de um método de prova para tal lógica. Na construção do Sistema Provedor de Teoremas, objeto do presente trabalho, foram utilizadas a Lógica Clássica de Primeira Ordem, conforme encontrada em [Enderton 2000], para representação do conhecimento; e o Método dos Tableaux, conforme encontrado em [Reves 1983] e [Buchsbaum & Pequeno 1990], para realização da prova automática de teoremas.

A atual etapa de desenvolvimento do Sistema Provedor de Teoremas, projeto de pesquisa em desenvolvimento no ULBRA/ILES de Ji-Paraná, enfatiza o estudo e desenvolvimento de uma interface que proporcione facilidades para utilização do sistema. Essa interface integrará quatro subsistemas: Sistema de Tableaux Tradicional, Primeiro, Segundo e Terceiro Refinamentos. Esses subsistemas foram projetados e implementados em [Fendt 2000], onde cada refinamento apresenta acréscimos ao Sistema Tradicional, com o intuito de evitar explosão do espaço de busca e obtenção de respostas para um número maior de problemas, já que a prova de teoremas é um problema indecidível.

A construção de uma interface gráfica para o sistema é feita para facilitar a sua compreensão pelo usuário principiante em Lógica, de modo que o mesmo, além de expressar uma forma de automatização do raciocínio, seja um eficiente instrumento didático para o estudante de Lógica e Automatização do Raciocínio.

2. Metodologia Utilizada para Desenvolvimento do Trabalho

Para a realização do proposto nesta etapa do trabalho, foi realizada inicialmente uma introdução a um ambiente de programação que utiliza a Linguagem de Programação LISP (do inglês *List Processing*), linguagem desenvolvida por John McCarthy em 1958. De modo a padronizar a interface com o ambiente Windows, foi utilizada uma versão de LISP do software *Allegro CL Integrated Development*. A família de ferramentas *Allegro CL* está baseada em Objetos Dinâmicos (programação orientada a objetos dinâmica), uma tecnologia via CLOS (*Common Lisp Object System*). CLOS é uma linguagem orientada a objetos dinâmica, comercialmente provada e não proprietária.

Paralelamente ao estudo do ambiente *Allegro CL* iniciou-se o projeto da interface. A metodologia utilizada para a construção da interface é a de Protótipos Descartáveis, conforme encontrada em [Carvalho 2001], onde são construídos e refinados protótipos até a obtenção de um que atenda os requisitos esperados. Cabe ressaltar que a interatividade dessa metodologia dispensa a construção de modelos dos protótipos intermediários, isto é, somente o protótipo final é modelado. Utilizou-se a lista de checagem Ergolist, encontrada em [Ergolist 2002], para verificação de conformidade do protótipo inicial com requisitos de usabilidade. As listas de checagem compõem uma técnica de avaliação rápida, destinadas a apoiar a inspeção da interface e descobrir defeitos ergonômicos mais flagrantes.

Após a conclusão do refinamento do protótipo da interface com Ergolist, esse deverá ser integrado com Sistema de Tableaux Tradicional e seus Refinamentos, chamados, de agora em diante, de Proveedor, através da elaboração de funções de mapeamento no sentido direto e no sentido inverso.

A função de mapeamento no sentido direto é responsável pela transformação das sentenças de entrada fornecidas pelo usuário ao sistema em sentenças manipuláveis pelo Proveedor. Essa função de mapeamento é necessária, pois o Proveedor utiliza uma representação interna que facilita o procedimento de prova de teoremas em detrimento da compreensão do usuário.

Em contrapartida, durante o procedimento de prova, é necessária uma função de mapeamento no sentido inverso, isto é, que mapeie a representação interna utilizada pelo Proveedor a uma representação externa, compreensível ao usuário. Essa função de mapeamento possibilita que o usuário possa acompanhar e compreender o processo da prova de teoremas passo a passo.

Após a integração do protótipo inicial da interface com o Proveedor, proceder-se-ão a testes de aceitação por usuários iniciantes em lógica (alunos das disciplinas de Lógica de Predicados) para o refinamento final da interface, verificando o cumprimento dos objetivos previstos para esta etapa do trabalho.

3. Protótipo inicial da Interface

O protótipo inicial da interface, como pode ser observado na **Figura 1**, foi elaborado seguindo o padrão de janelas do Sistema Operacional Windows. Nesse ambiente, o mouse é o instrumento de indicação mais utilizado. Todas as funções dos menus, submenus e escolha dos botões são executadas com o mouse; para isto o usuário movimenta o cursor (comumente representado por uma seta) na área de trabalho (tela).

O Proveedor possui uma rotina que controla o laço de execução do algoritmo e recebe três parâmetros. O primeiro parâmetro é o algoritmo a ser utilizado no processo de prova, que é uma alternativa entre o Sistema de Tableaux Tradicional, o Primeiro Refinamento, o Segundo Refinamento e o Terceiro Refinamento, todos encontrados em [Fendt 2000], o segundo é uma lista das fórmulas da base de conhecimento e a negação do possível teorema, e finalmente o terceiro parâmetro é o número máximo de nós que a árvore de prova deve alcançar.

A janela base, denominada “Prova de Teorema”, é responsável pela entrada dos parâmetros, descritos acima, necessários para execução do Sistema Proveedor de Teoremas. Os elementos desta janela são:

O Cabeçalho ou Barra de Título - contém o nome da janela base e o botão de menu da janela, onde se encontram as opções no menu: restaurar, mover, dimensionar, minimizar, maximizar e fechar, sendo esse último relativo ao encerramento do aplicativo.

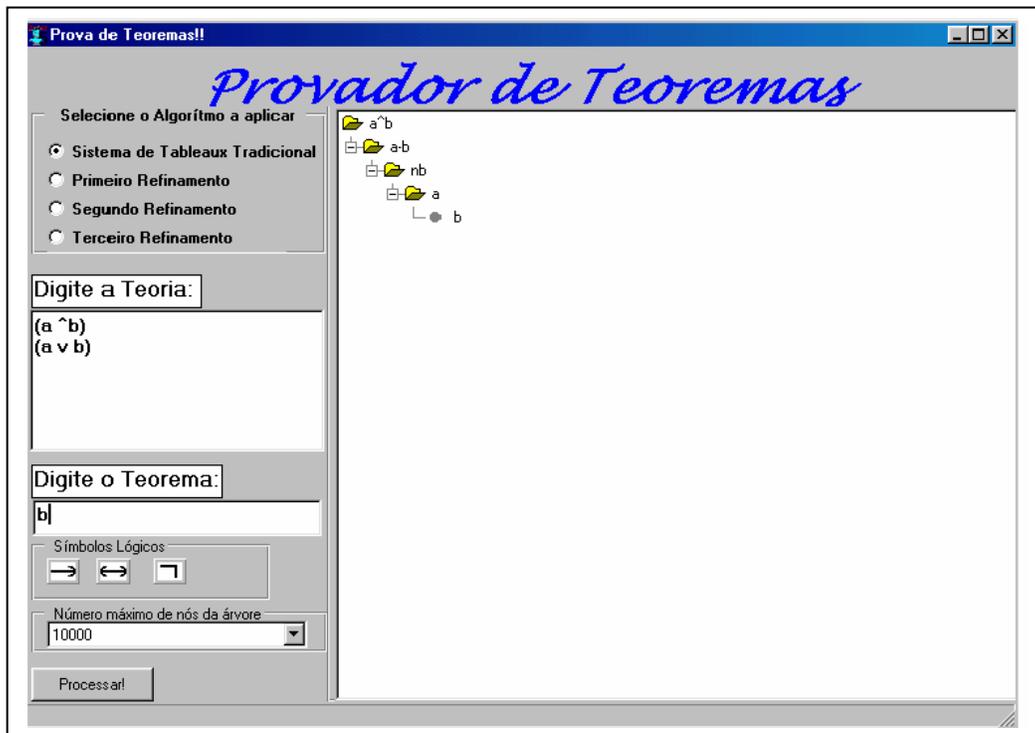


Figura 1: Protótipo Inicial da Interface do Sistema Provador de Teoremas

Selecione o Algoritmo a Aplicar – é um menu de escolha composto por botões rádio, para seleção do algoritmo (Tradicional, Primeiro, Segundo e Terceiro Refinamentos) a ser aplicado sobre os dados de entrada. O algoritmo default é o Sistema de Tableaux Tradicional.

Digite Teorema e Digite Teoria – são caixas de texto que solicitam informações ao usuário, e devem ser preenchidas com as fórmulas que compõem a teoria e o possível teorema que se deseja provar. O preenchimento desses campos é feito a partir do teclado, com o auxílio dos botões que contém símbolos lógicos, mostrados logo abaixo da entrada de dados do teorema.

Número Máximo de Nós da Árvore – é um componente do tipo combo-box, que permite a seleção do número máximo de nós que a árvore de prova pode atingir. Este campo, que possui valor default 10000, é de grande importância para que o sistema não execute até esgotar a memória da máquina, pois a prova de teoremas é um problema indecidível.

Processar – é um Botão de Comando, que indica a execução da operação de prova do teorema, com base nos parâmetros digitados e selecionados. Ao ser acionado, permanece sombreado até o final da operação.

Árvore Gerada – área de desenho, onde a árvore de prova construída pelo sistema pode ser visualizada. Nessa área as informações são apresentadas na forma de pastas, o que permite ao usuário um controle maior da visualização dos nós gerados. Os itens sinalizados com “•” representam as folhas da árvore de prova.

4. Checagem do Protótipo Inicial da Interface

O protótipo inicial da interface descrito no item anterior foi submetido a uma lista de checagem, de modo a permitir seu refinamento. A lista de checagem Ergolist, encontrada em [Ergolist 2002], é baseada no estudo de Dominique

Scapin, realizado com o intuito de organizar e facilitar o acesso aos conhecimentos de ergonomia de interfaces, por parte de especialistas e não especialistas.

Após a aplicação da lista de checagem foi possível observar os aspectos descritos a seguir, para cada um dos oito itens que compõem a lista.

Quanto à prestação, o sistema está carente na parte de informação quanto ao formato dos dados de entrada, ou ajuda direta ao usuário. Isto pode ser solucionado com a criação de um item de ajuda, descrevendo as funcionalidades do sistema, diferenças dos algoritmos e formatos de dados utilizados pelo sistema.

Já nos critérios agrupamento por itens e agrupamento por formato, o protótipo inicial interface apresentou uma avaliação favorável, indicando que seus elementos estão bem localizados e com características gráficas particulares a cada classe.

Outra carência do protótipo inicial da interface diz respeito às respostas imediatas da interface ao usuário, já que a mesma deixa de interagir com o usuário após a entrada dos dados, sem informar se o sistema está processando com sucesso ou não. Para sanar essa carência, o rodapé da janela pode ser utilizado para exibir mensagens sobre o funcionamento do sistema.

Na tentativa de contextualizar todo o processo sendo realizado em uma única tela o critério de legibilidade ficou prejudicado, devido à quantidade de informações, caixas de texto, botões e menus em uma única janela. Isto prejudica a avaliação quanto ao critério densidade informacional, pois dados que possuem valores default, com o número máximo de nós da árvore de prova, são questionados no protótipo inicial da interface. Uma opção é esconder do usuário as opções menos significativas, deixando a interface mais legível e com menos densidade informacional.

Já nos critérios concisão e ações mínimas o protótipo inicial da interface apresentou um bom desempenho, pois a interface foi projetada para exigir, com ações mínimas do usuário, somente os dados necessários para a prova do teorema, isto é, somente a teoria de base ou o possível teorema são campos obrigatórios. Valores default são fornecidos para entradas particulares deste sistema.

Nos critérios experiência do usuário e significados dos códigos e denominações o sistema recebeu uma avaliação razoável, já que foi desenvolvido para um usuário particular, com conhecimentos mínimos em Lógica de Predicados, porém mesmo para esse usuário é necessário um maior esclarecimento dos significados dos códigos e denominações. Estes critérios também evidenciam a necessidade de criação de um item de ajuda.

O protótipo inicial da interface não apresenta flexibilidade, isto é, o usuário não pode personalizá-lo. É possível atender a esse critério concedendo ao usuário a possibilidade de alterar valores default, por exemplo.

Outro fator importante da interface, identificado através da lista de checagem, foi a ausência de controle do sistema após a entrada de dados do usuário e o início do processamento do Provedor. É importante acrescentar uma forma do usuário interromper / reiniciar o funcionamento do sistema, ou mesmo um botão para processamento passo a passo.

Nos itens relacionados ao critério consistência que puderam ser avaliados, o protótipo inicial da interface apresentou um bom desempenho, pois esse item está relacionado a sistemas que apresentam uma diversidade maior de telas, menus, botões, etc.

Quanto à compatibilidade, buscou-se conservar o padrão Windows, para manipulações relacionadas a janelas e cores. A saída de dados, na forma de árvore de pastas, também é compatível com o método aplicado na sua forma tradicional da árvore de prova.

Outro ponto deficiente identificado no protótipo foi a ausência de tratamento para eventuais erros de entrada e processamento. Esse critério deverá ser tratado durante o desenvolvimento das funções de mapeamento no sentido direto, onde os dados são adequados ao funcionamento do sistema. Essas funções deverão validar os dados digitados diretamente pelo usuário.

5. Conclusão

O presente trabalho descreve a atual fase de desenvolvimento do Sistema Provedor de Teoremas, na qual a ênfase está no desenvolvimento de uma interface adequada para a utilização por iniciantes na área de Lógica. A interface descrita neste trabalho, desenvolvida através da metodologia de protótipos descartáveis, foi avaliada através da ferramenta Ergolist, o que possibilitou a identificação de uma série de pontos a serem sanados, para a sua adequação aos requisitos de usabilidade. Os principais pontos identificados foram a necessidade de adição de um item de ajuda para utilização do sistema, verificação de erros de entrada de dados, implementação de mensagens de erro, além da necessidade do usuário possuir um controle maior do sistema durante o seu processamento. Após esses ajustes serem implementados no protótipo inicial da interface, eles serão integrados ao Provedor e submetidos à aceitação dos usuários, ocasionando possíveis futuros refinamentos.

6. Referências Bibliográficas

[Buchsbaum & Pequeno 1990] Buchsbaum, Arthur & Pequeno Tarcisio, *O Método dos Tableaux Generalizado e sua Aplicação ao Raciocínio Automático em Lógicas Não Clássicas*. Revista “O que nos faz pensar” – Cadernos do Departamento de Filosofia da PUC-Rio, nº 3, setembro de 1990.

[Carvalho 2001] Carvalho, Adriana Maria Brito Rizzoni, *Introdução à Engenharia de Software*. Editora da UNICAMP, 2001.

[Enderton 2000] Enderton, Herbert B., *A Mathematical Introduction to Logic*. Academic Press, 2000.

[Ergolist 2002] Lista de Checagem Ergolist. Em <http://www.labiutil.inf.ufsc.br/ergolist>. Labutil – Laboratório de Usabilidade, Universidade federal de Santa Catarina, consultada em 15/07/2002, última atualização em 25/08/1998.

[Fendt 2001] Fendt, Letícia Carvalho Pivetta & Buchsbaum, Arthur, *O Método dos Tableaux com Unificação*. Encontro Nacional de Inteligência Artificial – XXI Congresso da SBC, 2001, Fortaleza, Anais do XXI Congresso da Sociedade Brasileira de Computação, 2001, v.1, p.185.

[Fendt 2000] Fendt, Letícia Carvalho Pivetta, *Refinamentos para o Método dos Tableaux*. Dissertação de Mestrado, Universidade Federal de Santa Catarina, Florianópolis, 2000.

[Nolt & Rohatyn 1991] Nolt, John & Rohatyn, Dennis, *Lógica*. McGraw-Hill & Makron Books, 1991.

[Reeves 1983] Reeves, S. V., *An Introduction to Semantic Tableaux*. Department of Computer Science, University of Essex, 1983.