

HEMERA – PROVADOR DE TEOREMAS

Aluno: Victor Hugo Marques
Orientador: Edward Hermann Haeusler

Motivação e Introdução

A Prova Automática de Teoremas lida com o desenvolvimento de programas de computador que demonstram que alguma sentença (a conjectura) é uma consequência lógica de um conjunto de sentenças (os axiomas e as hipóteses). Existem vários métodos para realizar provas de teoremas, com método Dedução Natural[2], Cálculo de Sequentes e Tableau[2]. Os métodos baseados em tableau são particularmente interessantes por prover de maneira óbvia um procedimento de decisão, sua fácil aplicação no ensino da lógica, e por poder ser utilizado para desenvolver procedimentos de prova para vários outros tipos de lógicas, clássicas ou não clássicas.

Objetivo

Este projeto objetiva dar continuidade no desenvolvimento do HEMERA[1], implementado pelo Prof Alexandre Rademaker (PUC-RIO), que atualmente implementa um provador de teoremas de lógica proposicional utilizando o método de Cálculo de Sequentes[2], introduzindo o método de prova por tableau. Para isso foi realizada a adaptação da sua máquina de inferência, para que realize provas utilizando o método de tableau, e foi também incluído um método para exibição gráfica de cada passo da prova.

Metodologia

Inicialmente, foram estudados todos os componentes básicos da linguagem proposicional, ou seja, seu alfabeto, os elementos da linguagem (que são chamados de fórmulas), sua semântica, os conceitos de satisfatibilidade e validade e consequência lógica [2].

Após este estudo básico lógica proposicional, foi feito um estudo do método de tableau analítico, que é um método de inferência baseado em refutação: para provarmos que $A_1, \dots, A_m \vdash B_1, \dots, B_m$, afirmaremos a veracidade de B_1, \dots, B_m , e a falsidade de A_1, \dots, A_m , na esperança de conseguirmos uma contradição. Se a contradição for obtida, teremos demonstrado o sequente.

E por fim, para implementação, foi estudada a linguagem de programação Python. Para isso foi realizada a leitura de um livro de referência da linguagem. Após adquirido conhecimento necessário da sintaxe e comandos da linguagem, foi feito um estudo completo dos módulos que já estavam implementados no provador, para obter um maior conhecimento sobre o funcionamento do provador no estado em que ele se encontrava.

Arquitetura

O provador para tableau é composto principalmente pelos módulos: PLY (Python Lex-Yacc), que é uma implementação das ferramentas de parse Lex[3] (que lê um fluxo de entrada especificando um analisador que mapeia expressões regulares em blocos de código) e YAAC[3] (um gerador de analisador sintático) em python; PyGraphviz[4], que é usado para criar, editar, ler e escrever grafos; Graph_Tableaux, que permite criar um grafo para representar a árvore de prova do tableau; e o módulo Tableaux que implementa as regras de inferência, as regras de expansão do tableau, armazena uma pilha dos ramos abertos, e utiliza

os módulos PLY, para realizar a verificação da entrada, e PyGraphviz, para gerar a representação gráfica da prova.

Desenvolvimento

Após o estudo dos módulos já presentes no provador de teoremas, foi iniciado o desenvolvimento dos módulos necessários para implementação da prova pelo método de tableau.

Primeiramente, foi implementado o módulo Graph_Tableaux, que contém estrutura do grafo utilizado para representar a árvore de prova, e métodos básicos para manipulação do grafo e foi realizado um teste para verificar a consistência dos algoritmos criados.

Posteriormente a criação do módulo Graph_Tableaux foi definido o módulo Tableaux, que recebe como entrada um sequente $A_1, \dots, A_m \vdash B_1, \dots, B_m$, valida essa entrada, e executa o algoritmo de prova, que pode ser resumido da seguinte forma:

Entrada: um sequente $A_1, \dots, A_m \vdash B_1, \dots, B_m$

Saída: Árvore de prova em formato DOT[4]

- 1- Criar um ramo inicial contendo $A_1, \dots, A_m \vdash B_1, \dots, B_m$,
- 2- **enquanto** existir uma ramo aberto **faça**
- 3- Escolher um ramo aberto.
- 4- Encontrar a fórmula do ramo R com menor custo.
- 5- Expandir o ramo R aplicando uma regra a fórmula escolhida
- 6- **fim enquanto**

Conclusão

A utilização do método tableau para provar teoremas de lógica proposicional criando uma representação gráfica da prova, por prover um sistema de inferência claro e que não necessariamente gera provas de tamanho exponencial, mostrou-se interessante em aplicações didáticas no ensino da lógica, pois é possível identificar todo o processo de prova e verificar qual a melhor maneira de provar determinado teorema.

Como trabalho futuro, pretendemos implementar uma prova para Lógica de Primeira Ordem, utilizando o método tableau.

Referências e Bibliografia

- [1] - RADEMAKER, Alexandre. HAEUSLER, Edward Hermann. Projeto HEMERA
- [2] - SILVA, Flávio Soares Corrêa da. **Lógica Para Comutação**. 1.ed Thomson Learning, 2006, 225 p.
- [3] - PLY (Python Lex-Yacc) - Documentação
<http://www.dabeaz.com/ply/>
- [4] - GRAPHVIZ – Documentação
<http://networkx.lanl.gov/pygraphviz/>