

# Geração de fórmulas da Lógica Proposicional

Filipe da Silva Oliveira<sup>1</sup>

Elthon Oliveira<sup>2</sup>

Ciência da Computação/UFAL, Arapiraca, AL

## 1 Introdução

MOOCs - *Massive Open Online Courses* são cursos *online* abertos e acessíveis a qualquer pessoa com acesso à internet [1]. Esses ambientes geralmente possuem uma grande quantidade de estudantes. Por isso, torna-se inviável a criação de exercícios únicos para cada aluno. Neste trabalho é apresentada uma abordagem de geração automática de fórmulas da Lógica Matemática Proposicional. Para isso, é utilizada uma combinação de especificações matemáticas com uma técnica de Síntese de Programas chamada Geração por Esboço [3]. Síntese de Programas consiste em gerar automaticamente um programa que satisfaça certas especificações. Na Geração por Esboço, é fornecida a uma linguagem subjacente a um esboço de um programa juntamente com uma especificação. A partir disso, é obtido o melhor programa que satisfaça ao esboço e à especificação.

Por programa, neste contexto, entenda-se *fórmulas bem formadas* - *fbf's* da Lógica Proposicional. Pode-se definir formalmente uma *fbf* a partir dos três casos bases que seguem: (i) toda letra do alfabeto é uma *fbf*, conhecido como átomo; (ii) se  $\alpha$  é uma *fbf*,  $\neg(\alpha)$  também é; e (iii) se  $\alpha$  e  $\beta$  são *fbf's*,  $\alpha$  o  $\beta$  também é, onde o corresponde a alguma das operações:  $\wedge$  |  $\vee$  |  $\rightarrow$  |  $\leftrightarrow$ .

As especificações matemáticas usadas para definir *fbf* são escritas utilizando a linguagem de modelagem Alloy [2]. A Alloy é uma linguagem baseada em lógica de primeira ordem capaz de descrever estruturas e relações. Há um suporte ferramental chamado *Alloy Analyzer*<sup>3</sup>, que é um solucionador que analisa estruturas, relações, propriedades e restrições em busca de modelos que as satisfaçam.

## 2 Especificações e Algoritmo

Na Figura 1 é apresentado parte do código Alloy que especifica formalmente uma *fbf*. O código ilustrado na Figura 1(a) especifica que há fórmulas unárias e binárias. Na Figura 1(b) é especificado que uma *fbf* pode ser: átomo, negação, conjunção, disjunção, implicação ou biimplicação. Já na Figura 1(c), é especificado para o processo de geração que uma fórmula tem como referência um operador principal. Como exemplo, na geração da fórmula  $C \wedge (A \vee B)$ , a conjunção é o operador a partir do qual a fórmula é gerada.

O algoritmo para geração das fórmulas é brevemente descrito a seguir. É fornecido ao *Alloy Analyzer* toda a especificação contendo os elementos, estruturas, propriedades e restrições sobre o que pode ser uma *fbf*. O *Alloy Analyzer* é integrado a outra ferramenta que recebe do usuário humano informações sobre características que as fórmulas geradas devem ter em comum. A partir

<sup>1</sup>filipe.oliveira@arapiraca.ufal.br - Bolsista PIBIC/CNPq-Af do Edital Propep/UFAL 2020-2021.

<sup>2</sup>elthon@arapiraca.ufal.br

<sup>3</sup><https://alloytools.org/>

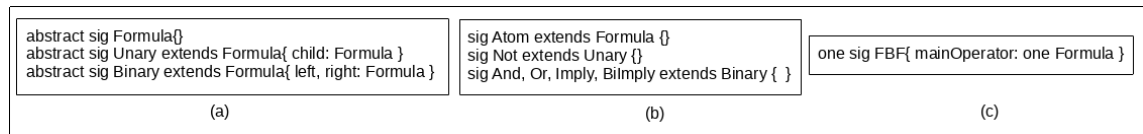


Figura 1: Trechos da especificação em Alloy para *fbf*.

da entrada fornecida pelo usuário e das especificações em Alloy, o analisador retorna diversos modelos que serão instanciados na forma de *fbf*. Na Figura 2 é ilustrado parte deste processo. Na Figura 2(a) é apresentado trecho de código Alloy gerado a partir das entradas fornecidas pelo usuário. Neste caso, o usuário escolheu gerar *fbf*'s que não tenham implicação ou biimplicação e que tenham entre 3 e 6 átomos, inclusive. Na Figura 2(b) é apresentado um dos vários modelos gerados a partir do código apresentado no passo anterior. Por último, na Figura 2(c), é apresentada a *fbf* instanciada a partir do modelo do passo anterior. As *fbf*'s são instanciadas por meio de uma API (*Interface de programação de Aplicações*, em inglês) desenvolvida em Java que acessa e utiliza a API da ferramenta *Alloy Analyzer*.

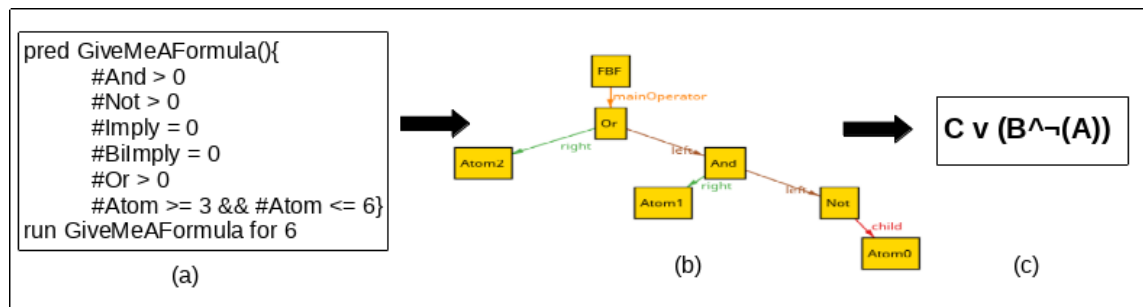


Figura 2: Passos na geração de uma *fbf*.

### 3 Resultados e conclusões

A abordagem apresentada sintetiza *fbf*'s com características similares a partir de especificações formais fornecidas. Agora é possível gerar questões únicas e com mesmo nível de dificuldade para ser utilizadas em ambientes de MOOCs, inicialmente para a disciplina de Lógica Matemática. Por fim, é importante ressaltar que existe um aplicativo móvel em desenvolvimento que poderá ser utilizado em qualquer celular com sistema Android.

### Referências

- [1] Fassbinder, A., Delmiro, M. E. e Barbosa, E. F. Construção e Uso de MOOCs: Uma Revisão Sistemática, *TEMA*, 25:332–341, 2014. DOI: <http://dx.doi.org/10.5753/cbie.sbie.2014.332>.
- [2] Jackson, D. *Software Abstractions: logic, language, and analysis*. MIT press, 2012.
- [3] Solar-Lezama, A. *Program synthesis by sketching*. University of California, Berkeley, 2008.