

Formalização e Prova de algoritmos de menor caminho usando Coq

João Vitor Fröhlich

Universidade do Estado de Santa Catarina
joaovitorfrohlich@gmail.com

Orientadora: Dra Karina Girardi Roggia

16/06/2023



Sumário

- 1 Introdução
- 2 Objetivos
- 3 Teoria de Grafos
- 4 Algoritmos
- 5 Coq
- 6 Implementação
- 7 Conclusões Parciais
- 8 Referências

- Formalização e Prova
 - Especificação Formal
- Algoritmos de menor caminho
 - Teoria de Grafos
- Coq
 - Assistentes de Provas

O objetivo geral deste trabalho é formalizar e provar em Coq algoritmos de busca do menor caminho entre dois pontos em grafos.

Objetivos Específicos

- 1 Estudar os principais algoritmos determinísticos de busca do menor caminho de grafos
- 2 Estudar os principais algoritmos heurísticos de busca do menor caminho em grafos
- 3 Implementar alguns algoritmos de busca do menor caminho em assistente de provas, que serão escolhidos de acordo com critérios a serem estabelecidos
- 4 Provar a corretude da implementação dos algoritmos definidos

- $G = \langle V, E, \delta_0, \delta_1 \rangle$;
 - Restrição de laço: $\forall e \in E, \delta_0(e) \neq \delta_1(e)$
 - Restrição de aresta paralela:
 $\forall e_1, e_2 \in E, \delta_0(e_1) = \delta_0(e_2) \wedge \delta_1(e_1) = \delta_1(e_2) \implies e_1 = e_2$
- Vizinhança:
 $\exists e \in E, (\delta_0(e) = u \wedge \delta_1(e) = v) \vee (\delta_0(e) = v \wedge \delta_1(e) = u)$
- Diretamente alcançável: $\delta_0(e) = u \wedge \delta_1(e) = v$

Exemplo - Grafo direcionado

- $G = \langle V, E, \delta_0, \delta_1 \rangle$;
- $V = \{v_1, v_2, v_3, v_4\}$;
- $E = \{e_1, e_2, e_3, e_4, e_5\}$;

	δ_0	δ_1
e1	v1	v2
e2	v1	v3
e3	v1	v4
e4	v2	v3
e5	v3	v2

Tabela: Definição de δ_0 e δ_1 no grafo de exemplo

Exemplo - Grafo direcionado

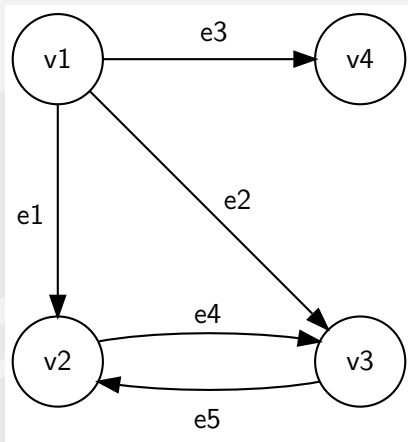


Figura: Grafo direcionado

Fonte: O autor

Definições e Exemplo - Grafo direcionado ponderado

- Adição de uma função $\varphi : E \rightarrow \mathbb{R}^+$
- $G = \langle V, E, \delta_0, \delta_1, \varphi \rangle$;

	φ
e1	2.5
e2	1.0
e3	5.0
e4	5.0
e5	0.5

Tabela: Definição de φ no exemplo 2

Exemplo - Grafo direcionado

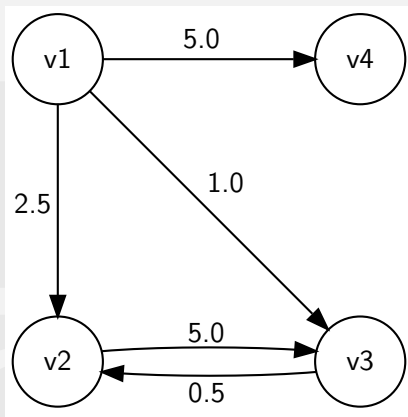


Figura: Grafo direcionado ponderado

Fonte: O autor

- Uma lista não vazia de arestas: $C = [e_1, e_2, \dots, e_n]$;
 - $\forall i \in \{1, n-1\}, \delta_1(e_i) = \delta_0(e_{i+1})$
- As funções δ_0 , δ_1 e φ (em grafos ponderados) podem ser definidas para cada caminho finito C no grafo, onde
 - $\delta_0(C) = \delta_0(e_1)$
 - $\delta_1(C) = \delta_1(e_n)$
 - $\varphi(C) = \sum_{i=1}^{|C|} \varphi(e_i)$
- Ciclos: $\delta_0(C) = \delta_1(C)$
- Menor caminho de u para v : C' tal que
 $\forall C (\delta_0(C) = u \wedge \delta_1(C) = v), \varphi(C') = \min(\varphi(C))$

Extensão do conceito de lógica modais com apenas uma (ou um par de) modalidade(s) que contém diversas modalidades. A linguagem de uma lógica multimodal é o menor conjunto LM_n que respeita:

- 1 $\top, \perp \in LM_n$
- 2 $\mathbb{P} \subseteq LM_n$
- 3 Se $\varphi \in LM_n$, então $\circ \varphi \in LM_n$, sendo $\circ \in \{\Box_1, \dots, \Box_n, \Diamond_1, \dots, \Diamond_n, \neg\}$
- 4 Se $\varphi, \psi \in LM_n$, então $\varphi \circ \psi \in LM_n$, sendo $\circ \in \{\wedge, \vee, \rightarrow\}$

- Assistente de provas para lógica de alta ordem, capaz de descrever e raciocinar sobre objetos matemáticos (GEUVERS, 2009);

- Assistente de provas para lógica de alta ordem, capaz de descrever e raciocinar sobre objetos matemáticos (GEUVERS, 2009);

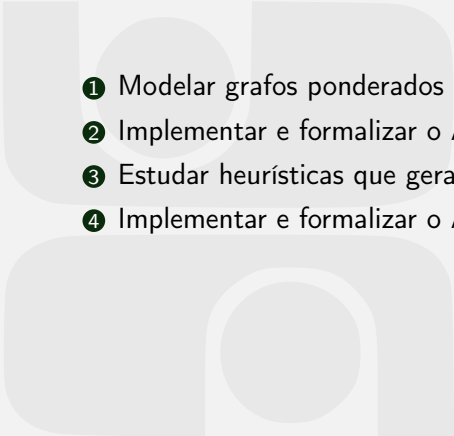
- Assistente de provas para lógica de alta ordem, capaz de descrever e raciocinar sobre objetos matemáticos (GEUVERS, 2009);
- O Coq é baseado em teoria de tipos, devido a Correspondência de Curry-Howard é capaz de expressar sistemas lógicos sofisticados;
- Coq tem uma grande quantidade de ferramentas de automação de provas, e também permite que seus usuários desenvolvam suas próprias ferramentas;
- Essas características tornam Coq uma boa ferramenta para representar sistemas lógicos complexos e operar sobre eles.

- Assistente de provas para lógica de alta ordem, capaz de descrever e raciocinar sobre objetos matemáticos (GEUVERS, 2009);


```
Inductive KT4formula : Set :=  
| T4Lit      : nat          -> KT4formula  
| T4Neg      : KT4formula -> KT4formula  
| T4Box      : KT4formula -> KT4formula  
| T4Dia      : KT4formula -> KT4formula  
| K4Box      : KT4formula -> KT4formula  
| K4Dia      : KT4formula -> KT4formula  
| T4And      : KT4formula -> KT4formula -> KT4formula  
| T4Or       : KT4formula -> KT4formula -> KT4formula  
| T4Implies  : KT4formula -> KT4formula -> KT4formula.
```

Figura: Linguagem de $\mathbf{KT} \odot \mathbf{K4}$

- A modelagem de grafos no Coq ainda está sendo estudada
- Resultados obtidos da implementação são positivos e indicam que é possível uma modelagem de grafos ponderados em Coq.

- 
- 1 Modelar grafos ponderados no Coq;
 - 2 Implementar e formalizar o Algoritmo de Dijkstra;
 - 3 Estudar heurísticas que geram o menor caminho na Busca A*;
 - 4 Implementar e formalizar o Algoritmo de Busca A*.

Etapas	2023/1	2023/2				
	Jul	Ago	Set	Out	Nov	Dez
1						
2						
3						
4						

Tabela: Cronograma Proposto para o TCC2



GEUVERS, H. Proof assistants: History, ideas and future.
Sadhana, Springer, v. 34, p. 3–25, 2009.