

Algoritmos em Grafos

Cláudia Linhares Sales

Julho 2020

Elementos das Buscas em Grafos

Serão vistos dois algoritmos de busca (ou visita) em grafos: **busca em largura** e **busca em profundidade**.

Dado um grafo (ou digrafo) $G = (V, E)$, os seguintes elementos são comuns aos dois algoritmos:

1. Cada vértice v tem pelo menos 2 atributos: sua **cor** e seu **pai**, que é o vértice pelo qual v foi descoberto;
2. As cores possíveis são 3: **branca**, indicando que o vértice nunca foi visitado; **preta**, indicando que o vértice já foi descoberto e a visita a partir de v já foi encerrada, ou seja que os vizinhos de v também já foram descobertos e **cinza**, que indica que o vértice já foi descoberto, mas eventualmente tem vizinhos não descobertos;
3. Tendo um vértice u sido descoberto, se um vizinho **branco** v de u é descoberto percorrendo a aresta (u, v) , então u passa a ser o pai de v ;
4. Ao final da execução do algoritmo, uma árvore ou floresta de busca G_π , que contém os vértices visitados, e permite reconstruir os caminhos pelos quais os vértices foram descobertos, mais especificamente:
 - $V(G_\pi) = \{v | cor[v] = preta\}$
 - $E(G_\pi) = \{(u, v) | pai[v] = u\}$

No caso da **Busca em Largura**, há mais elementos:

1. A visita sempre começa a partir de um vértice $s \in V(G)$;
2. Cada vértice v de G possui um atributo a mais: a **distância** de v , $d[v]$, que contém um limite superior para a distância (tamanho do menor caminho) entre s e v .

No caso da **Busca em Profundidade**, cada vértice v de G possui dois atributos a mais: o **tempo inicial** e **tempo final** de sua visita:

1. $b[v]$ indica o "tempo" ou passo em que o vértice v foi descoberto, passando de branco a cinza;
2. $f[v]$ indica o "tempo" ou passo em que o vértice v teve sua visita encerrada, passando de cinza a preto.

Cada algoritmo visita os vértices segundo uma ordem:

- A ordem da *Busca em Largura* pode ser assim resumida: tendo descoberto um vértice, visite todos os seus vizinhos não visitados (ou seja, brancos), antes de visitar os vizinhos de seus vizinhos. Em seguida, visite os vizinhos dos vizinhos, aplicando o mesmo princípio de busca.
- A ordem da *Busca em Profundidade* pode ser assim resumida, tendo descoberto um vértice, visite um vizinho não visitado (ou seja, branco). Caso todos os vizinhos tenham sido visitados, retroceda até o vértice mais recentemente visitado que ainda tenha vizinhos não visitados, e continue a aplicar o mesmo princípio de busca.

Finalmente, nos algoritmos apresentados, além de implementarem as ordens de visitas diferentes, há mais um aspecto que os difere:

- O Algoritmo de *Busca em Largura* que veremos visita **apenas** os vértices alcançáveis a partir de s , ou seja, os vértices tais que há um caminho entre eles e s . Logo G_π é uma árvore.
- O Algoritmo de *Busca em Profundidade* que veremos visita **todos** os vértices do grafo.