

Problema do Maior Caminho

Cléber Machado
João Luiz Grave Gross

4 de junho, 2012

1 Caracterização do problema

O problema do caminho mais longo é o problema de encontrar o caminho simples de maior comprimento em um grafo, ou seja, dentre todos os possíveis caminhos simples presentes em um grafo o problema é encontrar o maior. [URUY] Um caminho é chamado de simples se ele não contiver vértices repetidos.

Ao contrário do problema do menor caminho, onde busca-se encontrar o menor caminho entre dois vértices e pode ser solucionado em tempo polinomial para grafos sem ciclos de pesos negativos, este problema de decisão é NP-completo, logo não possui solução ótima em tempo polinomial a não ser que $P = NP$.

A versão de decisão padrão do problema do maior caminho em grafos procura encontrar um caminho simples de comprimento maior ou igual a k , onde o comprimento do caminho é definido pelo número de arestas ao longo do caminho. Para um grafo não valorado é suficiente encontrar o caminho mais longo, que tenha a maior quantidade de arestas e não repita vértices. Para grafos com arestas valoradas é preciso considerar seu peso ao invés da quantidade. [KHM] O foco deste artigo serão grafos com arestas não valoradas.

2 Prova que pertence a NP

A NP-completude do problema de decisão pode ser provada através da redução do problema do caminho hamiltoniano (que é conhecidamente NP-completo, veja [6]). Além disso, é necessário provar que o problema está em NP. Mas isto é trivial, pois um certificado para a instância sim do problema é a descrição de um caminho de comprimento maior ou igual a k .

2.1 Algoritmo de verificação

A versão deste problema que pode ser chamada de problema de decisão é a seguinte:

Problema Caminho (k, G): Dado um grafo G e um natural k , decidir se existe em G um caminho de comprimento maior ou igual a k .

Se houvesse um algoritmo polinomial para resolver este problema de

decisão, então haveria um algoritmo polinomial para resolver o problema do caminho mais longo (e, como provaremos a seguir, para resolver todos os problemas NP-completos). Suponha que a função caminho(k, G) receba um inteiro k e um grafo G e devolva sim se existe um caminho de comprimento maior ou igual a k em G e devolva não, caso contrário. Se esta função fosse polinomial, então o seguinte algoritmo, que encontra um caminho de comprimento máximo em um grafo G , seria polinomial.

```

CAMINHO MAIS LONGO( $G$ )
1   $k \leftarrow \|V(G)\| - 1$ 
2  enquanto (CAMINHO( $k, G$ ) = não)
3       $k \leftarrow k - 1$ 
4   $H \leftarrow G$ 
5  para  $i$  em  $E(H)$ 
6      remova  $i$  de  $H$ 
7      se (CAMINHO( $k, H$ ) = não)
8          insira  $i$  de volta em  $H$ 
9  devolva  $H$ 

```

Proposição 1 *O problema Caminho(k, G) é NP-completo.*

2.2 Análise da complexidade

3 Prova que pertence a NP-completo

Um grafo $G = (V, n)$, onde V é um grafo com n vértices. Definimos $f(<G>) = <G, n - 1>$, onde n é o número de vértices em G . Claramente f é computável em tempo polinomial: um algoritmo para computar f simplesmente conta o número de vértices do grafo G e anexa o valor decrescido de uma unidade ao seu resultado (saída), $n - 1$.

Assumimos que G possui um caminho hamiltoniano, ou seja, possui um caminho que contempla todos os vértices do grafo. Aplicamos a redução:

Caminho-Hamiltoniano \leq_p Caminho-Mais-Longo

Se $<G> \in$ Caminho-Hamiltoniano, então G tem um caminho hamiltoniano, i. e. G possui um caminho simples de comprimento $n - 1$, implica que para $f(<G>) = <G, n - 1> \in$ Caminho-Mais-Longo.

Analogamente, se $f(<G>) = <G, n - 1> \in$ Caminho-Mais-Longo, então G tem um caminho simples com $n - 1$ arestas, e também com n vértices. Como um caminho simples com n vértices é um caminho hamiltoniano, confirmamos que $<G> \in$ Caminho-Hamiltoniano.

Logo, mostrou-se que f é computável em tempo polinomial, e para todos os grafos G , $<G> \in$ Caminho-Hamiltoniano $\iff f(<G>) \in$ Caminho-Mais-Longo. Assim Caminho-Hamiltoniano \leq_p Caminho-Mais-Longo. Como Caminho-Hamiltoniano-Existe é NP-completo também podemos afirmar que Caminho-Mais-Longo é NP-completo. [MCP]

3.1 Redução $inst\ a \rightarrow inst\ b$

A redução do problema foi apresentada no item 3. Definimos que um grafo G é composto de V vértices e n arestas. Também definimos uma função f que recebe como parâmetro um grafo G e tem como saída o grafo com o tamanho do maior caminho, ou seja, $f(\langle G \rangle) = (G, n - 1)$.

Assumimos em G a existência de um caminho hamiltoniano e aplicamos G a f . O resultado foi um caminho simples de comprimento $n - 1$. Como a solução de f retorna apenas caminhos simples e sabendo que o caminho encontrado é hamiltoniano, comprovamos a existência do maior caminho simples em G , de tamanho $n - 1$.

3.2 Algoritmo de redução

A algoritmo de redução consiste em avaliar o grafo G , procurando a existência de um caminho hamiltoniano (`Caminho-Hamiltoniado-Existe()`). Caso G seja hamiltoniano, f retorna a quantidade de vértices de G decrescida de uma unidade (sempre considerando um grafo conexo). Eis o algoritmo:

`ReduceHamiltonianToLongestPath<G>`

1. inicializa a variável `path` (armazena o tamanho do caminho simples hamiltoniano)
2. **se** `Caminho-Hamiltoniado-Existe(G)` for verdade
3. **então** `path` é igual a quantidade de vértices em $G - 1$
4. **senão** não faz nada e retorna uma mensagem indicando erro
5. retorna a tupla $\langle G, path \rangle$

3.3 Análise da complexidade

O algoritmo de redução é bastante simples de compreender. Inicialmente é inicializada uma das variáveis de saída, isto é feito em tempo fixo, ou seja, $O(1)$. Nas linhas de 2 a 4 ocorre a execução principal do algoritmo. Na linha 2 um algoritmo de complexidade polinomial verifica a existência de um ciclo hamiltoniano em G . Se houver um caminho hamiltoniano, é feita a contagem dos vértices de G , ou seja, são feitas n operações, $O(n)$ e mais um subtração $O(1)$, totalizando um passo com $n + 1$ operações. Na linha 4 é realizada uma operação constante $O(1)$. Considerando a complexidade pessimista, nas linhas 3 e 4 assumimos a complexidade $O(n)$. Por fim, os resultados são retornados, $O(1)$.

Logo, somando-se as complexidades de inicialização, execução e finalização, ficamos com uma complexidade final de $O(n^m)$, onde $m > 1$ e n^m corresponde à complexidade polinomial da função `Caminho-Hamiltoniado-Existe()`.

4 Conclusão

Com este trabalho conseguimos realizar a prova de que o problema do maior caminho em grafos é um problema NP-completo, pois não possui solução em tempo polinomial. As provas e reduções nos mostraram que o problema em questão apenas é verificável em tempo polinomial, considerando grafos genéricos.

Grafos direcionados acíclicos por exemplo, apresentam solução em tempo linear, através da utilização de programação dinâmica. Porém nosso foco contempla toda a gama de grafos, e portanto apenas a verificação se deu em tempo polinomial.

Referências

[KHM] Khan, Mumit. CSE 221: Longest path in a directed acyclic graph (DAG). April 10, 2011.

[URUY] Uehara, Ryuhei and Uno, Yushi. On Computing Longest Paths in Small Graph Classes. Department of Information Processing, School of Information Science, Japan Advanced Institute of Science and Technology (JAIST). Japan, July 28, 2005.

[MCP] McCabe, Paul. University of Toronto. Professor of CSC363, Computational Complexity and Computability on Spring 2005. (<http://www.cs.toronto.edu/~pmccabe/csc363-2005S/>)

Links:

1. http://en.wikipedia.org/wiki/Longest_path_problem => algoritmo que resolve o problema do maior caminho em um grafo para grafos direcionados acíclicos. Esse algoritmo resolve o problema em $O(n)$. Mas para grafos genéricos esse algoritmo não é válido.
2. <http://www.csee.wvu.edu/~ksmani/courses/sp09/cc/lecnotes/npcproofs.pdf> -> diz que a prova de que o caminho mais longo é NP-completo é feita a partir da redução do problema do caminho hamiltoniano
3. <http://www.shannarasite.org/kb/kbse57.html> => explica a redução do ciclo hamiltoniano para o caminho mais longo
4. Prova de que o caminho mais longo de um grafo é NP-completo: http://www.tcs.hut.fi/Studies/T-79.240/2001/solutions_5.ps.
5. Outra prova de que o caminho mais longo em um grafo é NP-completo: <http://www.cs.toronto.edu/~pmccabe/csc363-2005S/test2-soln.pdf>