

MO417 - Complexidade de Algoritmos

Ata do exercício 23.2-3

Aula do dia 26/05/2009

Redator: Émerson Muraro.

Enunciado:

A implementação de heap de Fibonacci do algoritmo de Prim é assintoticamente mais rápida que a implementação de heap binário para um grafo esparso $G = (V, E)$, onde $|E| = \Theta(V)$? E no caso de um grafo denso, onde $|E| = \Theta(V^2)$? De que modo $|E|$ e $|V|$ devem estar relacionados para que a implementação de heap de Fibonacci seja assintoticamente mais rápida que a implementação de heap binária?

Resolução:

Analisando o algoritmo de Prim, percebemos que a diferença no tempo de execução entre o algoritmo de Prim com heap binário e o com heap de Fibonacci é ocasionada pela operação DECREASE-KEY. Assim, para o Prim com heap binário temos $O(E \lg V)$, enquanto que o Prim com heap de Fibonacci é $O(E + V \lg V)$.

Caso 1: onde $|E| = \Theta(V)$

Prim c/ heap binário: $O(E \lg V) = O(V \lg V)$

Prim c/ heap Fibonacci: $O(E + V \lg V) = O(V + V \lg V) = O(V \lg V)$

Portanto possuem assintoticamente o mesmo tempo (ordem).

Caso 2: onde $|E| = \Theta(V^2)$

Prim c/ heap binário: $O(E \lg V) = O(V^2 \lg V)$

Prim c/ heap Fibonacci: $O(E + V \lg V) = O(V^2 + V \lg V) = O(V^2)$

Portanto o Prim c/ heap Fibonacci é assintoticamente mais rápido que o Prim c/ heap binário.

Caso 3: definir uma relação entre $|E|$ e $|V|$ de modo que Prim c/ heap Fibonacci seja assintoticamente mais rápido

Se $E = w(V)$, temos:

Multiplicando os dois lados da igualdade por $\lg V$, temos: $E \lg V = w(V \lg V)$, do qual podemos concluir que $E \lg V = w(E)$. Como $E \lg V$ é igual a $w(V \lg V)$ e $w(E)$ podemos afirmar que:

$$E \lg V = w(E + V \lg V)$$

Portanto temos que $E \lg V$ é assintoticamente maior que $E + V \lg V$, provando que $O(E + V \lg V)$ é assintoticamente mais rápido que $O(E \lg V)$, ou seja, que o Prim c/ heap de Fibonacci é mais rápido que o c/ heap binário, desde que $E = w(V)$.

Outra prova Seria partirmos também de $E = w(V)$ e realizarmos as seguintes substituições:

Prim c/ heap binário: $O(E \lg V) = O(w(V) \lg V)$

Prim c/ heap Fibonacci: $O(E + V \lg V) = O(w(V) + V \lg V)$

Como as duas funções são "O" podemos comparar diretamente os seus parâmetros. Comparando $w(V) \lg V$ com $w(V)$, concluímos que $w(V) \lg V$ é assintoticamente maior que $w(V)$. Comparando $w(V) \lg V$ com $V \lg V$, concluímos que $w(V) \lg V$ é assintoticamente maior que $V \lg V$. Portanto, se $w(V) \lg V$ é assintoticamente maior que cada um dos termos da soma ele também é maior que a soma. Com isso, concluímos que o Prim c/ heap de Fibonacci é mais rápido que o c/ heap binário, desde que $E = w(V)$.

Referências

[1] Cormen, T.H.; Leiserson, C.E.; Rivest, R.L.; Stein, C.; Algoritmos: Teoria e Prática. Tradução da 2ª edição americana, 2002