CLASSES DE COMPUTAÇÃO PARALELA E EXEMPLOS DE PROBLEMAS QUE ADMITEM ALGORITMOS PARALELOS RÁPIDOS

(Resumo Estendido)

Routo Terada

INST. DE MATEMÁTICA E ESTATÍSTICA
UNIVERSIDADE DE S. PAULO
ABRIL DE 1987

1. Introdução

A classe NC consiste de problemas que podem ser solucionados muito rapidamente (em tempo polinomial em log n), em paralelo, com um número praticável (polinomial) de processadores. Muitos problemas naturais em NC são conhecidos; neste artigo tentamos identificar subclasses importantes de NC e dar alguns exemplos interessantes em cada subclasse. Apresentamos o conceito de NC^1-redutibilidade e sua aplicação (um problema R é NC^1-redutível ao problema S se R pode ser solucionado com circuitos uniformes de profundidade logarítmica utilizando oráculos para S). Apresentamos problemas que são completos em relação a essa redutibilidade, nas várias subclasses de NC.

2. A família de circuitos uniformes

Um *circuito (booleano)* α com n entradas e m saídas é um grafo orientado acíclico com nós (chamados *gates*) rotulados da forma descrita a seguir. O circuito α possui k "nós de entrada" com grau de entrada zero, rotulados $x_1, \ldots x_k$ respectivamente. Todos os outros nós de grau de entrada zero são rotulados com 0 ou 1. Todos os nós com grau de entrada um são rotulados com $\neg$ (*not*). Todos os outros nós tem grau de entrada dois e são rotulados com $\blacktriangle$ (*and*) ou $\blacktriangledown$ (*or*). Exatamente m nós são de saída e rotulados com $y_1, \ldots, y_m$, respectivamente. Todo nó de entrada possui pelo menos um caminho dele mesmo para algum nó de saída. Denotamos por $c(\alpha)$, a complexidade de α, o número de nós de α, e por $d(\alpha)$, a profundidade de α, o comprimento do caminho mais longo de algum nó de entrada para algum nó de saída. O circuito α computa a função $f:\{0,1\}^n \rightarrow \{0,1\}^n$ na maneira clássica.

Em geral, estamos interessados em computar uma função $f=\langle f_n\rangle$ onde $f_n:\{0,1\}^{g(n)} \rightarrow \{0,1\}^{h(n)}$, e $g(n)$ é monotônica estritamente crescente e $g(n)=n^{O(1)}$ (i.e., $g(n)=O(n^c)$ para alguma constante c). (A função f será tratada como uma união sobre n de f_n.) Uma *família de circuitos* com comprimento de entrada g e de saída h é uma sequência $\langle\alpha_n\rangle$, onde α_n é um circuito com $g(n)$ entradas e $h(n)$ saídas. A família $\langle\alpha_n\rangle$ *computa* a função f sse α_n *computa* f_n para todo

Definição Um *problema* R (com parâmetros de comprimento g e h) é uma família $\langle R_n\rangle$ de relações binárias tais que R_n está contido em $\{0,1\}^{g(n)} \times \{0,1\}^{h(n)}$. Uma família de circuitos $\langle\alpha_n\rangle$ *resolve* o problema R sse a função $\langle f_n\rangle$ computada por $\langle\alpha_n\rangle$ *realiza* R no seguinte sentido: Para cada n e cada x em $\{0,1\}^{g(n)}$, se

$R_n(x,y)$ se verifica para algum y então $R_n(x, f_n(x))$ se verifica

Neste artigo nós nos restringimos à família de circuitos "uniformes"; i.e., famílias $\langle\alpha_n\rangle$ para as quais existe um algoritmo que, dado n, gera facilmente o n-ésimo circuito α_n

Definição. NC^k é o conjunto de todos os problemas R solucionáveis por uma família de circuitos uniformes $\langle\alpha_n\rangle$ com $c(\alpha_n)\ n^{O(1)}$ (i.e., $c(\alpha_n)$ é limitado superiormente por algum polinomio em n), e $d(\alpha_n)=O(\log^k n)$. NC - [união sobre k de NC^k].

3. A classe NC^1, e NC^1-redutibilidade

Lembremos que NC^1 consiste de todos os problemas solucionáveis por uma família de circuitos uniformes de profundidade $O(\log n)$, onde n é o número de bits de entrada (a restrição de tamanho polinomial é redundante nesse caso). Exemplos de funções em NC^1 são: soma ou produto de dois inteiros de n bits cada, soma de n inteiros de n bits cada, multiplicação inteira ou booleana de matrizes, e ordenação de n inteiros com n bits cada. Circuitos para essas funções estão em [Sa], [BCP], e [MP].

Referências bibliográficas

[BCP] Borodin, A., Cook, S.A., and Hoover, H.J.:"Log depth circuits for division and related problems", *Proc. 17th. IEEE FOCS*, 1984.

[MP] Muller, D.E., and Preparata, F.P.: "Bounds to complexities of networks for sorting and switching", *J. ACM* 22,2 (April 1975), 195-201.

[Sa] Savage, J.E. : "The complexity of Computing", J. Wiley, New York, 1976.