Programação Dinâmica

Multiplicação de Matrizes

Definição

- Programação dinâmica (PD) é um paradigma de programação que tem como objetivo reduzir o tempo de execução de um problema utilizando soluções ótimas a partir de subproblemas previamente calculados.

Passos em uma PD

- Dividir o problema em sub problemas;
- Computar os valores de uma solução de forma *bottom-up* e armazená-los (memorização);
- Construir a solução ótima para cada subproblema utilizado os valores computados.

O problema da Multiplicação de Matrizes

- O problema em essência é trivial, pois se multiplicarmos sequencialmente as <u>n</u> matrizes obteremos facilmente o resultado esperado. O maior desafio que reside neste problema é otimizar o desempenho do programa que o resolve. Ou seja, multiplicar as matrizes de modo (ordem ótima) que o número de multiplicações escalares seja mínimo.

O problema da Multiplicação de Matrizes

- Multiplicação de matrizes não é comutativa (em geral, A x B ≠ B x A), mas associativa, o que significa que A x (B x C) = (A x B) x C;

- Podemos observar que em geral, multiplicar uma matriz (m x n) por uma matriz (n x p) exige (m x n x p) operações de multiplicação, em uma aproximação suficiente.

Exemplo

Suponha que queiramos multiplicar quatro matrizes A x B x C x D, de dimensões 200 x 2, 2 x 30, 30 x 20 e 20 x 5, respectivamente.

Usando a fórmula *mnp*, vamos comparar várias maneiras diferentes de avaliar esta multiplicação

Exemplo

Organização dos parênteses	Computação do custo	Custo
A x ((B x C) x D)	2 x 30 x 20 + 2 x 20 x 5 + 200 x 2 x 5	3.400
(A x (B x C)) x D	2 x 30 x 20 + 200 x 2 x 20 + 200 x 20 x 5	29.200
(A x B) x (C xD)	200 x 2 x 30 + 30 x 20 x 5 + 200 x 30 x 5	45.000

Como podemos observar, a ordem da multiplicação faz uma grande diferença no tempo de execução final.

Solucionando o Problema

Como determinar a ordem ótima?

- Dividir o problema em subproblemas
- Seja:

  $C(i, j)$ = custo mínimo de multiplicar $M_i \times M_{i+1} \times \ldots \times M_j$

  - O tamanho de subproblemas é o número de multiplicação de matrizes, $|j - i|$.
  - O menor problema ocorre quando $i = j$, neste caso, $C(i, j) = 0$.

Solucionando o Problema

$$C(i, j) = \min \{C(i,k) + C(k+1, j) + m_{i-1}m_k m_j\},$$

onde k varia de 1 até (j – 1).

Neste caso, k varia de 1 a 3

$C(1, 4) = \min \{C(1,1) + C(2,4) + 200 \times 2 \times 30;\ C(1,2) + C(3,4) + 200 \times 30 \times 5;\ C(1,3) + C(4,4) + 200 \times 20 \times 5\}$

Solucionando o Problema

$C(1,1) = C(2,2) = C(3,3) = C(4,4) = 0$

$C(1,2) = 200 \times 2 \times 30 = 12.000$

$C(2,3) = 2 \times 30 \times 20 = 1.200$

$C(3,4) = 30 \times 20 \times 5 = 3.000$

$C(1,3) = \min\{C(1,1) + C(2,3) + 200 \times 2 \times 20;\ C(1,2) + C(3,3) + 200 \times 30 \times 20\} =$
$\min\{9.200;\ 132.000\} = 9.200$

Solucionando o Problema

C(2,4) = min{C(2,2) + C(3,4) + 2 x 30 x 5; C(2,3) + C(4,4) + 2 x 20 x 5}
min{3.300; 1.400} = 1.400

Portanto:

C(1, 4) = min {C(1,1) + C(2,4) + 200 x 2 x 30; C(1,2) + C(3,4) + 200 x 30 x 5; C(1,3) + C(4,4) + 200 x 20 x 5}

C(1,4) = min{3.400; 45.000; 29.200} = 3.400

Logo, a solução ótima é: $\mathbf{M_1 \times ((M_2 \times M_3) \times M_4)}$

Conclusões

- Tentar todas as ordens possíveis de multiplicações para avaliar o produto de n matrizes é um processo exponencial em n;
- Usando programação dinâmica é possível resolver este problema em $O(n^3)$