This and other notes are available from `http://www.cs.berkeley.edu/~yozo/cs170.fa05/`

Chain Matrix Multiply

Let A be a $m \times \ell$ matrix, B be a $\ell \times n$ matrix. Then a $m \times n$ matrix is the product $C = A \times B$ if

$$c_{ij} = \sum_{k=1}^{\ell} a_{ik} b_{kj}.$$

- Is matrix multiply associative (i.e., $(AB)C = A(BC)$)?
- Are they commutative (i.e., $AB = BA$)? Does this even make sense for all dimensions?
- What is the cost of (naïve) matrix multiplication? Note that there are faster methods (Strassen is $O(n^{2.81})$), but not numerically as desirable. Best is due to Winograd and Coppersmith with $O(n^{2.376})$, but not practical.
- Consider product of three matrices: ABC. Does the cost of matrix multiply matter if we do $(AB)C$ or $A(BC)$? Try for $A = 100 \times 2$, $B = 2 \times 200$, $C = 200 \times 3$.

Now we can ask the question: what is the best way to insert parenthesis to minimize the cost? Let $A_{i...j} = A_i A_{i+1} \cdots A_j$.

- Suppose the last step (of optimal sequence) was multiplying $A_{1...k}$ and $A_{k+1...n}$. How can we express the optimal cost in terms of optimal cost of the subproblem?
- Do we know what is the value of k? If not, how do we find out? Just try all of them!

$$C_{ij} = \min_{i \le k < j} (C_{i,k} + s_{i-1} s_k s_j + C_{k+1,j})$$

- What are the base cases?
- Write down the algorithm to compute the optimal cost. Running time? Best is $O(n \log n)$ due to Hu and Shing (1982).
- Show in which order the $n \times n$ table of C_{ij} can be filled in.
- Note that this can be generalized: there is no reason the operation is the matrix multiply. If an operation is associative, and the cost is given by $f(x, y)$ to combine x and y, then we can apply the same algorithm.

All-Pairs Shortest Paths

Given a weighted graph G, we want to find the shortest distance between all pairs $u, v \in V$.

- One possibility is to run Dijkstra or Bellman Ford $|V|$ times. What is the running time of this simple scheme?
- Let $D_{ij}^{(k)}$ be the shortest path from i to j using only intermediate nodes $1, 2, \ldots, k$.
- Write a recursion formula. What are the base cases?

- What is the running time?
- What is the required memory? Can you reduce this to $O(n^2)$? Why is this optimal (in asymptotic sense)?
- How do we construct the shortest paths?

More Questions

1. Consider chain matrix multiply for n matrices. What is the maximum ratio of worst solution to best solution?
2. What is the best all-pairs shortest paths algorithm you can come up with for unweighted graphs? Weighted graphs with non-negative edge weights? Consider both dense and sparse graphs.
3. For all-pairs shortest path problem, we consider the following method. Let $D_{ij}^{(k)}$ be the shortest path from i to j using at most k edges.
   - Write a recursion formula for $D_{ij}^{(k)}$.
   - What are the base cases?
   - What is the running time?
   - Is this any better than the Floyd-Warshall algorithm?
4. Why is the integer knapsack problem presented in class not considered polynomial?
5. In the integer knapsack problem presented in class, what happens if there are negative item sizes? Negative weights?