Design and Analysis of Algorithms (XXIII)

Conclusion

Guoqiang Li
School of Software

Shanghai Jiao Tong University

Algorithmic View of Algorithms

Algorithms Strategies

	D&C	DP	Greedy	Reduction	Duality
Basic topics	merge sort, median, matrix multiplication	LCS, edit distance, matrix chain multiplication	MST, Huffman encoding	graph/tree algorithm	max-flow/min-cut algorithms, maximal matching/vertex cover
Theoretical analysis	master theorem Akra-Bazzi theorem	time-space transformation, top-down, bottom-up	-	efficient/inefficient problems	primal-dual, simplex
Advanced topics	FFT	Sequence Alignment, Markov chain, treewidth	approximation algorithms	complexity classes, Karp reduction, Turing reduction	Lagrange duality, approximation algorithms

Algorithms on Data Structures

	Graphs	Flows	Numbers	...
Basic topics	DFS, BFS, DAG, shortest path, MST	Ford-Fulkerson algorithm	basic arithmetic, modular arithmetic	...
Theoretical analysis	priority queue, disjoint set, proof techniques	well-formed structures	expectation analysis, probability analysis,	...
Advanced topics	SCC, negative-cost path, GI	Dinitz algorithms, dynamic tree algorithms	RSA algorithm	...

Practical Algorithms

Q: Does Algorithm lecture provides practical, engineering algorithms?

Practical Algorithms

Q: Does Algorithm lecture provides practical, engineering algorithms?

A: Not really.

Practical Algorithms

SHANGHAI JIAO TONG UNIVERSITY

Q: Does Algorithm lecture provides practical, engineering algorithms?

A: Not really.

The terminology of algorithm in the *lecture* is quite different from the terminology of algorithm in *algorithm engineer*.

Algorithms Strategies

SHANGHAI JIAO TONG UNIVERSITY

	D&C	DP	Greedy	Reduction	Duality
Basic topics	merge sort, median, matrix multiplication	LCS, edit distance, matrix chain multiplication	MST, Huffman encoding	graph/tree algorithm	max-flow/min-cut algorithms, maximal matching/vertex cover
Theoretical analysis	master theorem Akra-Bazzi theorem	time-space transformation, top-down, bottom-up	-	efficient/inefficient problems	primal-dual, simplex
Advanced topics	FFT	Sequence Alignment, Markov chain, treewidth	approximation algorithms	complexity classes, Karp reduction, Turing reduction	Lagrange duality, approximation algorithms
Engineering/ Practical algorithms	Engineering FFT	time/space compression	Boruvka algorithm, clustering	DPLL/CDCL algorithms, invariant generation	deep learning explanation and verification

Algorithms on Data Structures

	Graphs	Flows	Numbers	. . .
Basic topics	DFS, BFS, DAG, shortest path, MST	Ford-Fulkerson algorithm	basic arithmetic, modular arithmetic	. . .
Theoretical analysis	priority queue, disjoint set, proof techniques	well-formed structures	expectation analysis, probability analysis,	. . .
Advanced topics	SCC, negative-cost path, GI	Dinitz algorithms, dynamic tree algorithms	RSA algorithm	. . .
Engineering/ Practical algorithms	Tarjan algotihm, algorithmic formal verification	push-relabelling algoithm	Miller-Rabin algorithm	. . .

Graphs and Flows

2. (10 points) Show that if an undirected graph with n vertices has k connected components, then it has at least n-k edges.

3. (10 points) Suppose someone presents you with a solution to a max-flow problem on some network. Give a linear time algorithm to determine whether the solution does indeed give a maximum flow.

8. (15 points) Suppose in a given network, all edges are undirected (or think of every edge as bi-directional with the same capacity), and the length of the longest simple path from the source s to sink t is at most L. Show that the running time of **Edmond-Karp algorithm** on such networks can be improved to be $O(L \cdot |E|^2)$.

Dynamic Programming

6. (15 points) Given two strings $x = x_1 x_2 \dots x_n$ and $y = y_1 y_2 \dots y_m$, we wish to find the length of their longest common substring, that is, the largest k for which there are indices i and j with $x_i x_{i+1} \dots x_{i+k-1} = y_j y_{j+1} \dots y_{j+k-1}$. Show how to do this in time $O(mn)$.

2. (10 points) Give an $O(n \cdot t)$ algorithm for the following task.
   Input: A list of n positive integers $a_1, a_2, \ldots a_n$, and a positive integer t.
   Question: Does some subset of the a_i's add up to t? (You can use each a_i at most once)

7. (15 points) A subsequence is **palindromic** if it is the same whether read left to right or right to left. For instance, the sequence

$$A, C, G, T, G, T, C, A, A, A, A, T, C, G$$

has many palindromic subsequences, including A, C, G, C, A and A, A, A, A. Devise an algorithm that takes a sequence $x[1, \ldots, n]$, and returns the (length of the) longest palindromic subsequence. Its running time should be $O(n^2)$.

5. (15 points) Consider the following **3-partition problem**. Given integers $a_1, \dots, a_n$, we want to determine whether it is possible to partition of $\{1, \dots n\}$ into three disjoint subsets I, J, K such that

$$\sum_{i \in I} a_i = \sum_{j \in J} a_j = \sum_{k \in K} a_k = \frac{1}{3} \sum_{i=1}^{n} a_i$$

Devise and analyze an algorithm for **3-partition** that runs in time polynomial in n and in $\sum_i a_i$.

Linear Programming

1. (10 points) Write the dual to the following linear program.

$$\text{Max } \ 6x - 4z - 3$$
$$3x - y \le 1$$
$$4y - z \le 2$$
$$x, y, z \ge 0$$

Is the solution $(x, y, z) = (1/2, 1/2, 0)$ optimal? Write the dual program of the given linear program and find out its optimal solution.

3. (10 points) In a **facility location problem**, there is a set of facilities and a set of cities, and our job is to choose a subset of facilities to open, and to connect every city to some one of the open facilities. There is a nonnegative cost f_j for opening facility j, and a nonnegative connection cost $c_{i,j}$ for connecting city i to facility j. Given these as input, we look for a solution that minimizes the total cost. Formulate this facility location problem as an integer programming problem.

1. (10 points) Write the dual to the following linear program.

$$\begin{aligned} \text{Max } & x/2 + 2y - 4 \\ & x + y \leq 3 \\ & x/2 + 3y \leq 5 \\ & x, y \geq 0 \end{aligned}$$

Find the optimal solutions to both primal and dual LPs.

5. (15 points) **Set Multicover problem** is defined as an extension of **Set Cover problem**, such that each element, e, needs to be covered a specified integer number, r_e, of times. The objective again is to cover all elements up to their coverage requirements at minimum cost. We will assume that the cost of picking a set S k times is $k \cdot cost(S)$. Represent the problem as an integer linear program. Then relax to a LP and work out its dual.

1. (10 points) In a **Feedback Vertex Set Problem** we are given a undirected graph $G = (V, E)$ and nonnegative weights $w_v > 0$ for $v \in V$, find a set $S \subseteq V$ of minimum weight such that $G[V \setminus S]$ is a forest. Formulate this facility location problem as an integer linear programming problem, relax it to an LP, and work out its dual.

3. (10 points) Find the dual of the following linear program.

$$
\begin{aligned}
max\ 6x_1 + 8x_2 + 5x_3 + 9x_4 + 5 \\
2x_1 + x_2 + x_3 + 3x_4 > 5 \\
x_1 + 3x_2 + x_3 + 2x_4 = 3 \\
x_1, x_2 \geq 0
\end{aligned}
$$

1. (10 points) Show that the following optimization problem can be modelled as an equivalent linear program:

$$min|x - y|, where\ x + y < 1$$

4. (10 points) **Steiner Forest Problem** is defined as follows,
Input an undirected graph $G = (V, E)$, nonnegative costs $c_e \geq 0$ for all edges $e \in E$ and k pairs of vertices $(s_i, t_i) \in V$.
Output a minimum cost subset of edges F such that every (s_i, t_i) pair is connected in the set of selected edges.
Represent the problem as an integer program.

NP Problem

4. (15 points) Given a reduction from the **Clique Problem** to the following problem, which you also need to show to be a search problem.

Input a undirected graph G and a positive integer k.

Output a Clique of size k as well as an Independent Set of size k, provided both exist.

2017

SHANGHAI JIAO TONG UNIVERSITY

4. (15 points) Prove that the **Graph-Isomorphism problem** is a NP problem.

7. (15 points) In the **Hitting Set** problem, we are given a family of sets $\{S_1, S_2, \ldots, S_n\}$ and a budget b, and we wish to find a set H of size $\leq b$ which intersects every S_i, if such an H exists. In other words, we want $H \cap S_i \neq \emptyset$ for all i.
Show that **Hitting Set** is NP-complete.

6. (15 points) Suppose you are helping to organize a summer sports camp, and the following problem comes up. The camp is supposed to have at least one counselor who's skilled at each of the n sports covered by the camp (baseball, volleyball, and so on). They have received job applications from m potential counselors. For each of the n sports, there is some subset of the m applicants qualified in that sport. The question is: For a given number $k < m$, is it possible to hire at most k of the counselors and have at least one counselor qualified in each of the n sports? We'll call this the **Efficient Recruiting Problem**.

Show that Efficient Recruiting is NP-complete.

7. (15 points) The **set packing problem** is defined as follows: Given a set *E*, a list $S_1, \ldots, S_m$ of subsets of *E* and an integer *k*, are there *k* sets in the list that are pairwise disjoint?
Prove that the **set packing problem** is a NP complete problem.

Approximation Algorithms

2015

SHANGHAI JIAO TONG UNIVERSITY

5. (15 points) You are given an undirected graph. The problem is to remove a minimum number of edges such that the residual graph contains no triangle. i.e., there is no three vertices a, b, c such that edges (a, b),(b, c),(c, a) are all in the residual graph. Give a factor 3 approximation algorithm that runs in polynomial time.

6. (15 points) A **Minimum Makespan Scheduling** problem is as follows:

Input processing times for n jobs, $p_1, p_2, \ldots, p_n$, and an integer m.

Output an assignment of the jobs to m identical machines so that the completion time is minimized.

Design an approximation algorithm by a greedy approach on the problem, with the approximation factor 2. And also give a tight example to show the approximation guarantee.

7. (10 points) The **Maximum Cut** problem is defined as follows: Given an undirected graph $G = (V, E)$ along with a nonnegative weight $w_{ij} \geq 0$ for each $(i, j) \in E$. The goal is to partition the vertex set into two parts, U and $W = V - U$, so as to maximize the weight of the edges whose two endpoints are in different parts. Give a 2-approximation randomized algorithm for maximum cut problem.

2. (15 points) Find a polynomial time 4/3-approximation algorithm for instance of **Metric TSP** where the distances are either 1 or 2.
   Hint: The 2-match problem (find a minimum weight subset of edges M such that each node is adjacent to exactly 2 edges in M) can be solved in polynomial time.

2017

3. (15 points) Consider the following algorithm for **Cardinality Vertex Cover**: In each connect component of the input graph execute a **depth first search (DFS)**. Output the nodes that are not leaves in the DFS tree. Show that the output is indeed vertex cover, and that it approximates the minimum vertex cover within a factor of 2.

5. (15 points) Consider the following two algorithms for the **knapsack problem**: (i) The greedy algorithm (pick the item with best value of $profit(i)/size(i)$); (ii) The algorithm that packs the maximum profit item.

Prove that the algorithm that picks the better of these two solutions is a 2-approximation for the knapsack problem.

6. (15 points) A **Minimum Makespan Scheduling** problem is as follows:

Input: processing times for n jobs, $p_1, p_2, \ldots, p_n$, and an integer m.

Output: an assignment of the jobs to m identical machines so that the completion time is minimized.

We can firstly decrease order of size and send each job to the shortest queue so far.

(a) Prove that this algorithm obtains a 3/2-approximation.

(b) Give the worst gap example you can think of for this algorithm.

2. (10 points) We know that by a greedy approximation algorithm on **Set Cover problem**, the approximation factor H_n is

$$1 + 1/2 + \cdots + 1/n$$

where n is number of elements in universal set. Give a tight example to show the approximation guarantee.

5. (15 points) Consider an optimization version of the **Hitting Set Problem** defined as follows. We are given a set $A = \{a_1, \ldots, a_n\}$ and a collection $B_1, B_2, \ldots, B_m$ of subsets of A. Also, each element $a_i \in A$ has a *weight* $w_i \geq 0$. The problem is to find a hitting set $H \subseteq A$ such that the total weight of the elements in H, that is, $\sum_{a_i \in H} w_i$, is as small as possible. Let $\mathrm{b} = \max_{\mathrm{i}} |\mathrm{B_i}|$ denote the maximum size of any of the sets $B_1, B_2, \ldots, B_m$. Give a polynomial-time approximation algorithm for this problem that finds a hitting set whose total weight is at most b times the minimum possible.

8. (15 points) Given a graph G and two vertex sets A and B, let $E(A, B)$ denote the set of edges with one endpoint in A and one endpoint in B.

The **max equal cut problem** is defined as follows: Given a graph $G(V, E)$, $V = \{1, 2, \ldots, 2n\}$, find a partition of V into two n-vertex sets A and B, maximizing the size of $E(A, B)$. Consider the following approximation algorithm for the **max equal cut problem**:

- Start with empty sets A, B, and perform n iterations.
- In iteration i, pick vertices $2i - 1$ and $2i$, and place one of them in A and the other in B, according to which choice maximizes $|E(A, B)|$. (i.e., if $|E(A \cup \{2i-1\}, B \cup \{2i\})| \geq |E(A \cup \{2i\}, B \cup \{2i-1\})|$ then add $2i - 1$ to A and $2i$ to B, else add $2i$ to A and $2i - 1$ to B.)

Prove that the algorithm has approximation factor is 1/2.