Weekplan: Approximation Algorithms I

Inge Li Gørtz

References and Reading

[1] Algorithm Design, Kleinberg and Tardos, Addison-Wesley, section 11.0, 11.1, 11.2. *Available on DTU Learn.*

[2] The Design of Approximation Algorithms, Williamson and Shmoys, Cambridge Press, section 2.2 + 2.3.

[3] A unified approach to approximation algorithms for bottleneck problems, D. S. Hochbaum and D. B. Shmoys, Journal of the ACM, Volume 33 Issue 3, 1986.

We expect you to read either [1] and [2] in detail. [3] provides background on the k-center problem.

Exercises

1 Acyclic Graph Given a directed graph $G = (V, E)$, pick a maximum cardinality set of edges from E such that the resulting graph is acyclic. Give a 1/2-approximation algorithm for this problem.

Hint: Arbitrarily number the vertices and pick the bigger of the two sets, the forward going edges and the backward going edges.

2 Minimum Maximal Matching A matching in a graph $G = (V, E)$ is a subset of edges $M \subseteq E$, such that no two edges in M share an endpoint. A maximal matching is a matching that cannot be extended, i.e., it is not possible to add an edge from $E \setminus M$ to M without violating the constraint.

Design a 2-approximation algorithm for finding a smallest maximal matching in an undirected graph, that is the maximal matching that has the smallest number of edges.

Hint: Use the fact that any maximal matching is at least half the largest maximal matching.

3 List scheduling: Refined analysis Show that the greedy list scheduling algorithm obtains an approximation factor of $2 - 1/m$.

4 Shipping consultant[1] You are a consultant for a large Danish shipping company "Ships, Ships, and Ships". They have the following problem. When a ship arrives at a port they have to unload the containers from the ship onto trucks. A ship carries containers with different weights, $w_1, w_2, \ldots, w_n$. Each truck can carry multiple containers, but only up to a total weight of W. The shipping company wants to use as few trucks as possible to unload the ship. This is a NP-complete problem.

You suggest that they use the following greedy algorithm: Consider the containers in any order. Start with an empty truck and begin stacking containers on it until you get to a container that would overload the truck. This truck is now declared loaded and sent away, and you continue with a new truck.

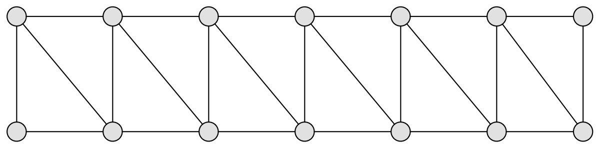
This algorithm might not be optimal, but it is simple and easy to implement in practice.

4.1 Prove that the number of trucks used by the algorithm is within a factor of 2 from the optimum.

4.2 Show that this is tight. That is, give an example, that shows that the algorithm might use (almost) twice as many trucks as the optimum solution.

[1] inspired by [1]

5 [w] k**-center** Run both k-center algorithms on the example below with $k = 4$. All edges have length 1.



6 The k-supplier problem The *k-supplier* problem is similar to the k-center problem, but the vertices are partitioned into *suppliers* $F \subseteq V$ and *customers* $C \subseteq V$. The goal is to find k suppliers such that the maximum distance from a customer to a supplier is minimized. Give a 3-approximation algorithm for the k-suppliers problem.

7 Metric k-clustering Give an 2-approximation algorithm for the following problem.

Let $G = (V, E)$ be a complete undirected graph with edge costs satisfying the triangle inequality, and let k be a positive integer. The problem is to partition V into sets $V_1, \ldots, V_k$ so as to minimize the costliest edge between two vertices in the same set, i.e., minimize

$$\max_{1 \leq i \leq k,\, u,v,\in V_i} c(u, v).$$

8 Longest processing time rule In this exercise we will show that LPT obtains an approximation factor of 4/3. Assume $t_1 \geq t_2 \geq \cdots \geq t_n$. You can assume wlog. that the smallest job finishes last.

8.1 Show that if $t_n \leq |T^*|/3$ then LPT gives a 4/3-approximation.

8.2 [$*$] Prove that for any input where the processing time of each job is more than a third of the optimal makespan, LPT computes an optimal schedule.

8.3 Use 1. and 2. to show that LPT is a 4/3-approximation algorithm.

9 Consultant You are a consultant for a company that has a number of servers and need to schedule batches of jobs. Once a batch of n jobs arrives they need to be allocated to servers. The company has two types of servers: k fast servers and m slow servers. Each job i takes time t_i to process on a slow server, and time $t_i/3$ to process on a fast server. The goal is to minimize the makespan of the schedule.

You suggest that they use the simple greedy algorithm: Process jobs in any order. Assign next job on list to machine with smallest current load.

9.1 Give an example showing that this algorithm is not a 3-approximation algorithm.

9.2 Prove that this is a 4-approximation algorithm.

9.3 Give a better approximation algorithm for the problem.