

## Lecture 1: Approximation Algorithms Review

*Lecturer: Allan Borodin**Scribe: Kevin Gao*

The conceptually simplest “combinatorial” algorithms:

- brute force search
- greedy algorithms
- dynamic programming
- divide and conquer
- local search

Consider the **knapsack** problem. We are given a set of  $n$  items  $I_1, \dots, I_n$  and a size bound  $B$  where each item  $I_j = (w_j, v_j)$  with  $w_j$  being the weight (cost) of the item and  $v_j$  the value of the item.

A subset of items  $S$  is feasible if the sum of the weights of items in  $S$  is at most  $B$ . The goal of the knapsack problem is to find a feasible set  $S$  that maximizes the sum of the values of items in  $S$ .

A “natural” algorithm for knapsack is to sort the items by their ratios of value over cost  $v_i/w_i$ . However, this only works for **fractional** values. 0/1 knapsack is **NP-complete**.

## 1.1 Greedy and Online Algorithms

We start with two greedy algorithms: Graham’s online and LPT makespan algorithms.

### 1.1.1 Makespan

In the **makespan** problem, we have  $m$  machines and  $n$  jobs  $\{J_1, \dots, J_n\}$ . Each job  $J_i$  has a load  $P_i$ . The goal of the problem is to assign every job to a machine. We want to minimize the maximum load on any machine. More formally, we want to find a mapping  $\sigma : \{J_1, \dots, J_n\} \rightarrow \{1, \dots, m\}$  from the set of jobs to the set of machines that **minimizes** the objective function

$$\max_j \left\{ \sum_{i: \sigma(i)=j} P_i \right\}$$

The natural online greedy algorithm for makespan achieves an approximation ratio of  $2 - \frac{1}{m}$ .

The analysis of asymptotic approximation ratio is commonly known as competitive analysis (a term coined by Manasse, McGeoch and Sleator). In **competitive analysis**, we compare the performance of an online algorithm against that of an optimal solution. An **online algorithm** is one where input items arrive sequentially and the algorithm must make an irrevocable decision concerning each item. For makespan, an item is a job and the decision is to choose a machine on which the item is scheduled. The competitive ratio is the standard measure of performance of online algorithms, although there are alternative measures.

An adversarial input for the greedy makespan algorithm would be  $m(m-1)$  job of load 1 and one job of load  $m$ . This results in a makespan of  $m-1 + m = 2m-1$ . The optimal solution has make span  $m$  (reserve one machine for the job with load  $m$ ).

A better (not online) approximation algorithm is to sort the jobs by their load. We call this LPT (longest processing time)-first. This gives an approximation ratio of  $\frac{4}{3} - \frac{1}{3m}$ .

### 1.1.2 Variants of Online Algorithms

In the standard meaning of online algorithms, we think of an adversary as creating an adversarial input set and the ordering of the input items in that set. We also call this online **adversarial input model**.

We will also consider an **online stochastic model**. In this model, an adversary defines an input distribution and then input items are sequentially generated. This is often seen in operational research.

In the **random order model** (ROM), an adversary creates a size  $n$  adversarial input set and then the items from that set are given in a uniform random order (uniform over the  $n!$  permutations).

There may be extensions to this paradigm, for example:

- Lookahead with buffer
- Limited ability to revoke previous decisions
- Multiple solutions and take the best of the final solutions
- Parallelism or multiple passes over the input items
- Advice bits based on the entire input