Greedy Algorithms

A Scheduling Application: The Algorithm

Algorithms: Design and Analysis, Part II

Intuition for Algorithm

Recall: want to min $\sum_{j=1}^{n} w_j C_j$

Goal: devise correct greedy algorithm

Question: (1) with equal lengths, schedule larger- or smaller-weight jobs earlier?

(2) with equal weights, schedule shorter or longer jobs earlier?

(A) larger / shorter

(B) smaller / shorter

(C) larger / longer

(D) smaller / longer

Resolving Conflicting Advice

Question: what if $w_i > w_j$ but $\ell_i > \ell_j$?

Idea: assign "scores" to jobs that are:
- increasing in weight
- decreasing in length

Guess ①: order jobs by decreasing value of $w_j - \ell_j$

Guess ②: order w_j / ℓ_j

Breaking A Greedy Algorithm

To distinguish ① & ②: find example where the two algorithms produce different outputs (at least one will be incorrect).

Example: larger ratio, larger difference

$\ell_1 = 5$, $w_1 = 3$ — $\ell_2 = 2$, $w_2 = 1$

Alg #1: #1, #2 — $1 \cdot 2 + 3 \cdot 7 = 23$

Alg #2: #2, #1 — $3 \cdot 5 + 1 \cdot 7 = 22$

Question: what is the sum of weighted completion times of algorithms ① & ②, respectively?

(A) 22 and 23
(B) 23 and 22
(C) 17 and 17
(D) 17 and 11

The Story So Far

So: Alg#1 not (always) correct.

Claim: Alg#2 (order by decreasing ratio w_j/ℓ_j's) is always correct.

[not obvious! — proof coming up next]

Running time: $O(n \log n)$ [just need to sort].