Algorithms: Design and Analysis, Part II

Greedy Algorithms

A Scheduling Application: Handling Ties

Correctness Claim

Claim: Algorithm #2 (order jobs in nonincreasing order of ratio w_j/l_j) is always correct. [even with ties]

New Proof Plan: fix arbitrary input of n jobs. Let σ = greedy schedule, let σ^* = any other schedule.

Will show σ at least as good as σ^* $\Rightarrow$ implies that greedy schedule is optimal.

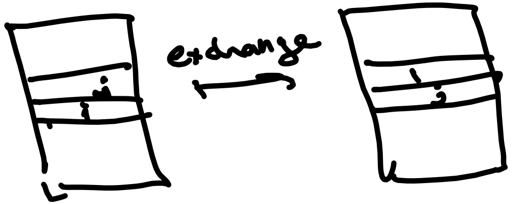
Correctness Proof

Assume: [just by renaming jobs] greedy schedule σ is just $1,2,3,\ldots,n$ (and so $w_1/\ell_1 \geq w_2/\ell_2 \geq \cdots \geq w_n/\ell_n$).

Consider arbitrary schedule σ^*. If $\sigma^* = \sigma$, done.
Else recall $\exists$ consecutive jobs i, j in σ^* with $i > j$. (from last time)

Note: $i > j \Rightarrow w_i/\ell_i \leq w_j/\ell_j \Rightarrow w_i \ell_j \leq w_j \ell_i$

Recall: exchanging i & j in σ^* has net benefit of $w_j \ell_i - w_i \ell_j \geq 0$



Correctness Proof (con'd)

Upshot: exchanging an "adjacent inversion" like i, j only makes σ^* better, and it decreases the number of inverted pairs.

jobs i, j with $i > j$ and i scheduled earlier

$\Rightarrow$ after at most $\binom{n}{2}$ such exchanges, can transform σ^* into σ

like BubbleSort!

$\Rightarrow$ σ at least as good as σ^*

$\Rightarrow$ greedy is optimal

QED!