



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 $\sigma^* \Rightarrow$ Implies that greedy schedule is optimal.

Correctness Proof

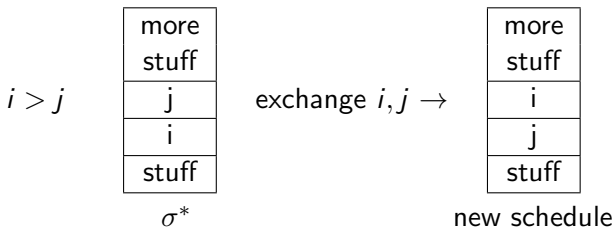
Assume: [Just by renaming jobs] Greedy schedule σ is just $1, 2, 3, \dots, n$ (and so $w_1/l_1 \geq w_2/l_2 \geq \dots \geq w_n/l_n$).

Consider arbitrary schedule σ^* . If $\sigma^* = \sigma$, done.

Else recall \exists consecutive jobs i, j in σ^* with $i > j$. (From last time)

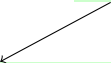
Note: $i > j \Rightarrow w_i/l_i \leq w_j/l_j \Rightarrow w_i l_j \leq w_j l_i$.

Recall: Exchanging i & j in σ^* has net benefit of $w_j l_i - w_i l_j \geq 0$.



Correctness Proof

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 .

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

\Rightarrow Greedy is optimal.

QED!