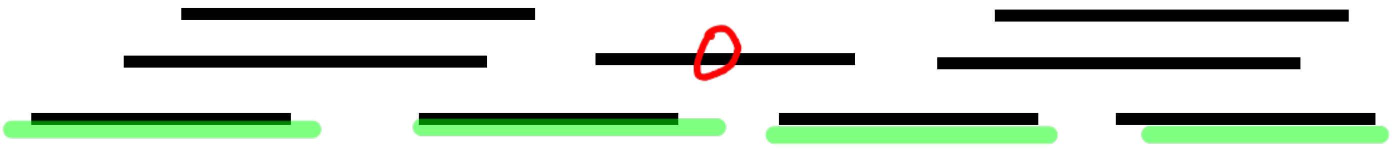
Greedy II: Interval scheduling.

The problem: given n intervals $[s(i), f(i)]$, select the maximum number that is disjoint.

Example: 

max 4

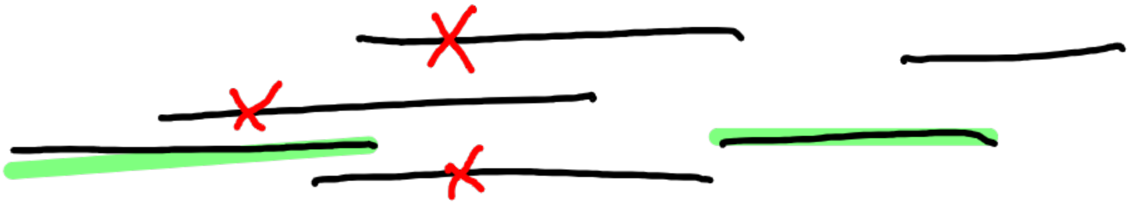
Last time we discussed greedy ideas, and were two last ones left

A. Select the one with earliest finish time

~~B.~~ Select the one with fewest conflict

not working

Algorithm A: select interval with earliest finish time



Algorithm:

While any intervals left

Select the one with earliest finish time, and accept it

delete all overlapping intervals

Endwhile

Running time:

n intervals on our list

$O(n \log_2 n)$ sort by finish time

at most n iterations

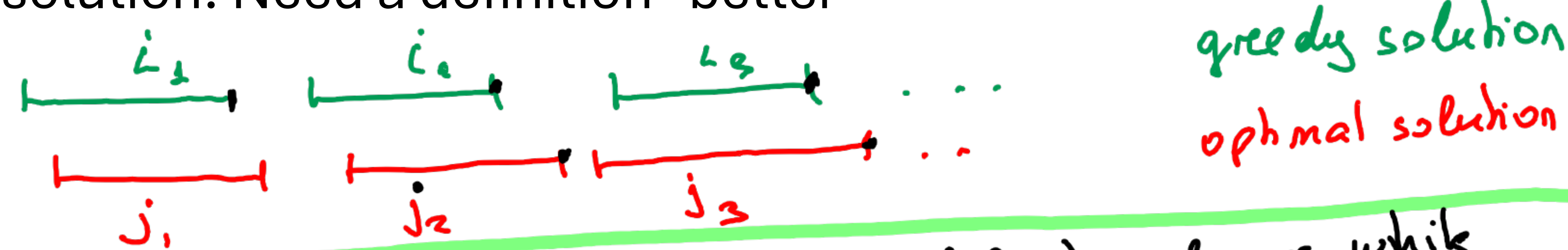
easy to implement each iteration in $O(n)$ time

$\Longrightarrow$ total time $O(n^2)$

also possible $O(n \log n)$

Proving Algorithm A correct:
proof technique: greedy stays ahead

Will prove by induction that at each step greedy is “better” than any other solution. Need a definition “better”



$\boxed{\text{Claim: all } r \quad f(i_r) \le f(j_r) \quad \text{for } r \text{ while Opt has an interval}}$

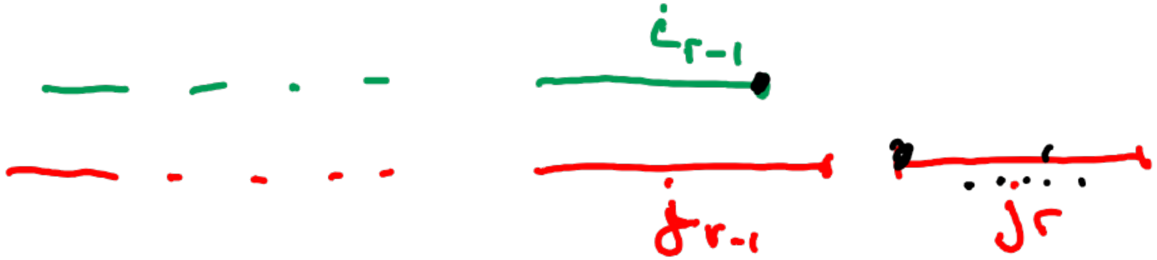
immidiately implies that greedy has at least as many intervals as Opt

$\Longrightarrow$ greedy Optimal

Proof of Claim by induction on r

<u>base case</u> $r=1$ by definition of greedy $f(i_1) \leq f(j_1)$

<u>induction step.</u> by induction hypothesis $f(i_{r-1}) \leq f(j_{r-1})$



$s(j_r) \geq f(j_{r-1})$
as they are disjoint

combining the two
$s(j_r) \geq f(i_{r-1}) \implies j_r$ is among intervals greedy considered when selecting its next interval

& greedy rule: select one with earliest finish time $\quad f(i_r) \leq f(j_r)$

✓