Algorithms: Design and Analysis, Part II

Dynamic Programming

WIS in Path Graphs: A Reconstruction Algorithm

Optimal Value vs. Optimal Solution

Recall: $A[0] = 0$, $A[1] = w_1$, for $i = 2, 3, 4$ to n
$A[i] := \max\{A[i-1], A[i-2] + w_i\}$

Note: algorithm computes the value of a max-wt IS, not such an IS itself.

Correct but not ideal: store optimal IS of each G_i in the array in addition to its value.

Better: trace back through filled-in array to reconstruct optimal solution.

Key point: we know that a vertex v_i belongs to a max-weight IS of G_i $\iff$ w_i + max-wt IS of G_{i-2} $\geq$ max-wt IS of G_{i-1}

follows from correctness of our algorithm!

Tim Roughgarden

A Reconstruction Algorithm

Let A = filled-in array A:

0	4	4	7	...	184
0	1	2	3	...	n

- Let $S = \emptyset$
- While $i \geq 1$ [scan through array from right to left]
  - if $A[i-1] \geq A[i-2] + w_i$: [i.e., Case 1 wins]
    - decrease i by 1
  - else [i.e., Case 2 wins]
    - add v_i to S, decrease i by 2
- return S

Claim: [by induction + our case analysis] final output S is a max-wt IS of G.

Running time: $O(n)$.