CS483 Analysis of Algorithms
Lecture 07 – Dynamic Programming 01

Jyh-Ming Lien

June 29, 2017

Dynamic Programming

▷ Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

- A term coined by Richard Bellman in the 1940s

(Image from ieee.org. Richard Bellman, 1920 - 1984)

- Some problems solved by dynamic programming
  - Longest increasing subsequences
  - Fibonacci number
  - Knapsack problem
  - All-pairs shortest path problem (Floyd's algorithm)
  - Optimal binary search tree problem
  - Multiplying a sequence of matrices
  - String matching (or DNA sequence matching), where we search for the string closest to the pattern
  - Convex decomposition of polygons
  - ...

Optimization

Dynamic Programming
▷ Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

- Things you need to know about dynamic programming (dp)
  - **programming** in DP (and linear programming) is a mathematical term, which means **optimization** or planning, i.e. it should not be confused with “computer programming” or “programming language”
  - dp solves problems with **overlapping sub-problems**
  - dp solves problems which have **optimal substructure**, i.e., its optimal solution can be constructed from optimal solutions of its sub-problems
  - dp stores the results of sub-problems for later reuse
  - dp works by converting a problem into a set of sub-problems and representing these sub-problems as a DAG.

Weighted Interval Scheduling

Dynamic Programming
Optimization
▷ Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

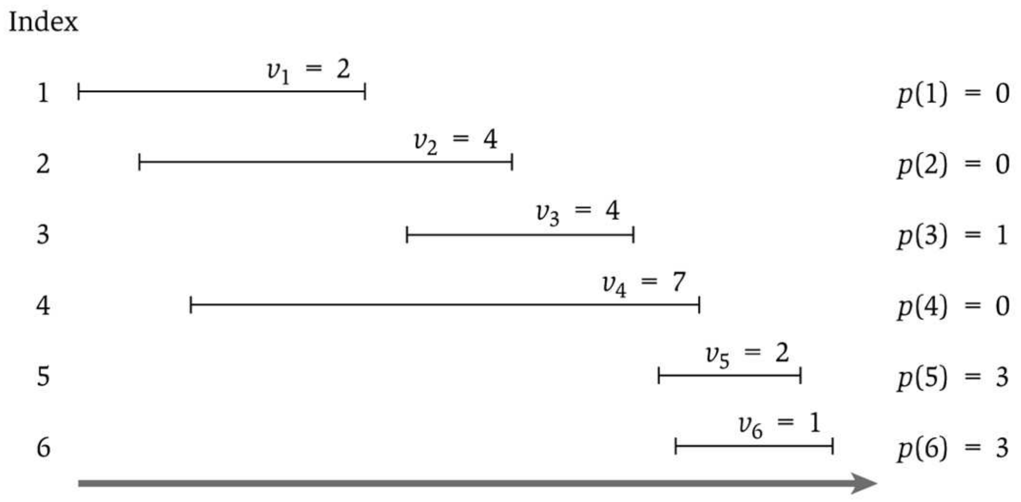


Figure 6.2 An instance of weighted interval scheduling with the functions $p(j)$ defined for each interval j.

Weighted Interval Scheduling

Dynamic Programming
Optimization
Weighted Interval Scheduling
▷ Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

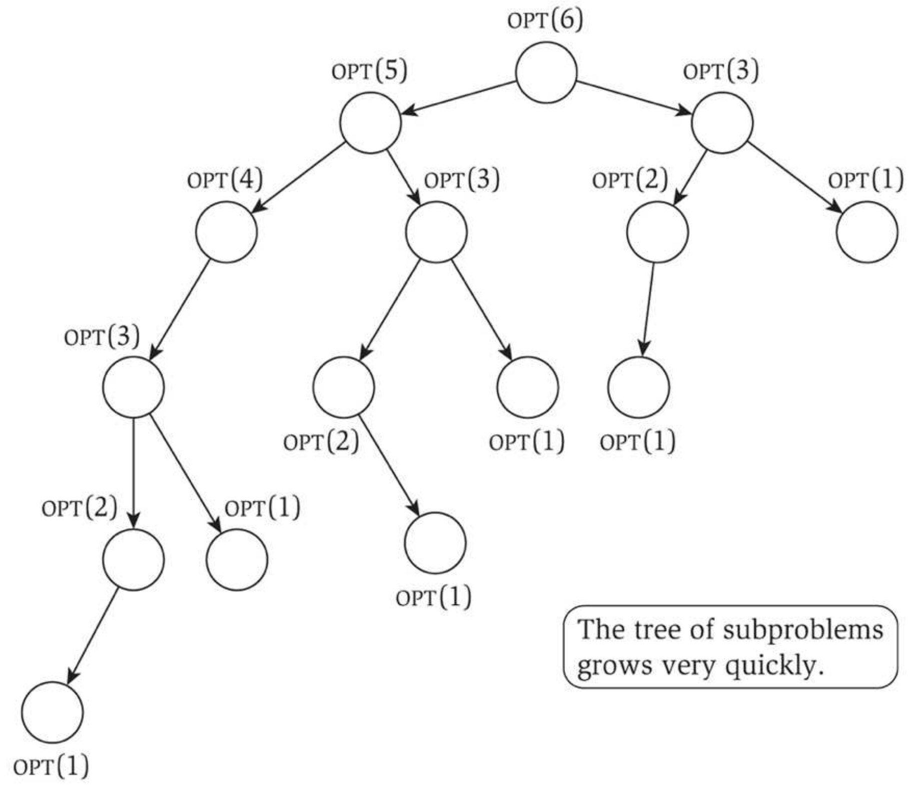
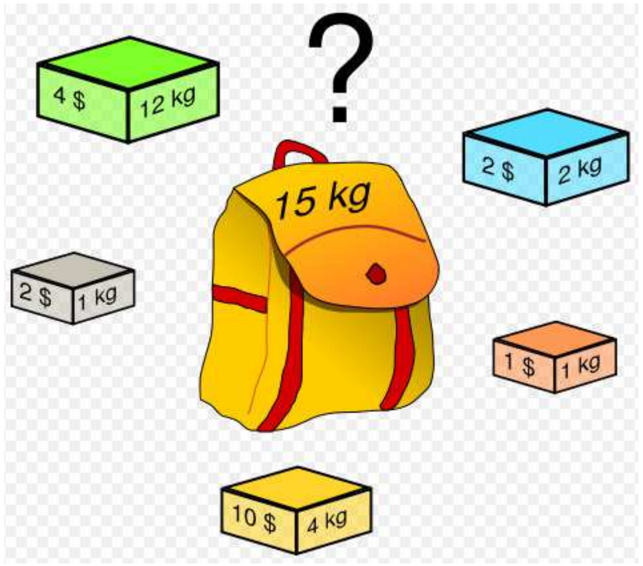


Figure 6.3 The tree of subproblems called by `Compute-Opt` on the problem instance of Figure 6.2.

Knapsack Problem

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
▷ Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

- Knapsack Problem: Given n objects, each object has weight w and value v, and a knapsack of capacity W, find most valuable items that fit into the knapsack



- Brute force approach
  - generate a list of all potential solutions
  - evaluate potential solutions one by one
  - when search ends, announce the solution(s) found
- What is the time complexity of the brute force algorithm?

Knapsack Problem

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
▷ Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

- Dynamic programming approach
  - Assume that we want to compute the optimal solution $S(w, i)$ for capacity $w < W$ with i items
  - Assume that we know the optimal solutions $S(w', i')$ for all $w' \leq w$ and $i' \leq i$
  - Option 1: **Don't add** the k-th item to the bag, then $S(w, i) = S(w, i-1)$
  - Option 2: **Add** the k-the item to the bag, then $S(w, i) = S(w - w_i, i-1) + v_i$

w	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
12kg, $4															
1kg, $2															
2kg, $2															
1kg, $1															
4kg, $10															

- Time complexity?

Knapsack Problem and Memory Functions

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
▷ Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

- So far, we look at four DP-based algorithms, all of them are bottom-up approaches.
- We can in fact design DP-based algorithms using top-down (recursive) approach.
  - One important benefit of top-down approach is that we can avoid solving unnecessary subproblems

w	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
12kg, \$4															
1kg, \$2															
2kg, \$2															
1kg, \$1															
4kg, \$10															

- Algorithm

```
Algorithm 0.1: NAPSK($w, i$)

if $V[w, i] < 0$
  then { if $w < W[i]$
           then $value \leftarrow$ NAPSK($w, i-1$)
           else $w < W[i]$
           then $value \leftarrow \max\{$NAPSK($w, i-1$),NAPSK($w - W[i], i-1) + V[i]\}$
         $V[w, i] \leftarrow value$
return $(V[w, i])$
```

Piecewise Linear Regression

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
▷ Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

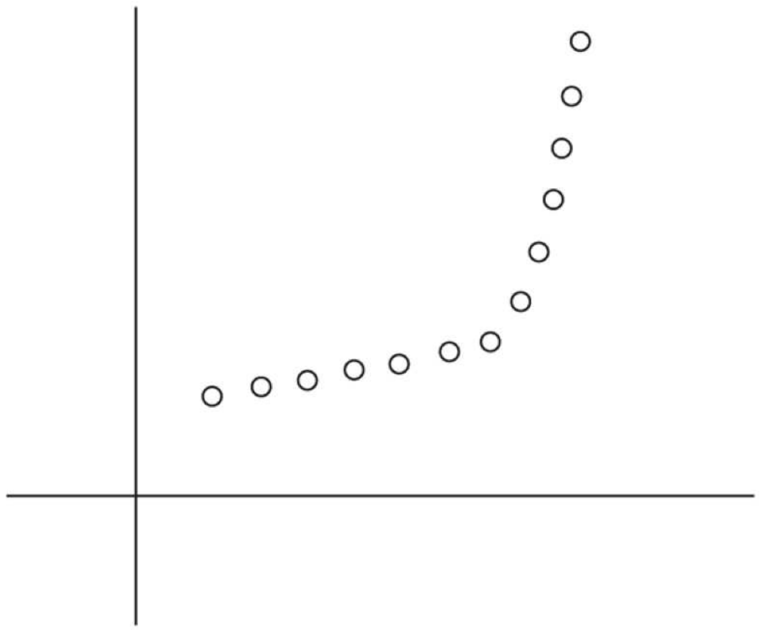


Figure 6.7 A set of points that lie approximately on two lines.

Piecewise Linear Regression

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
▷ Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

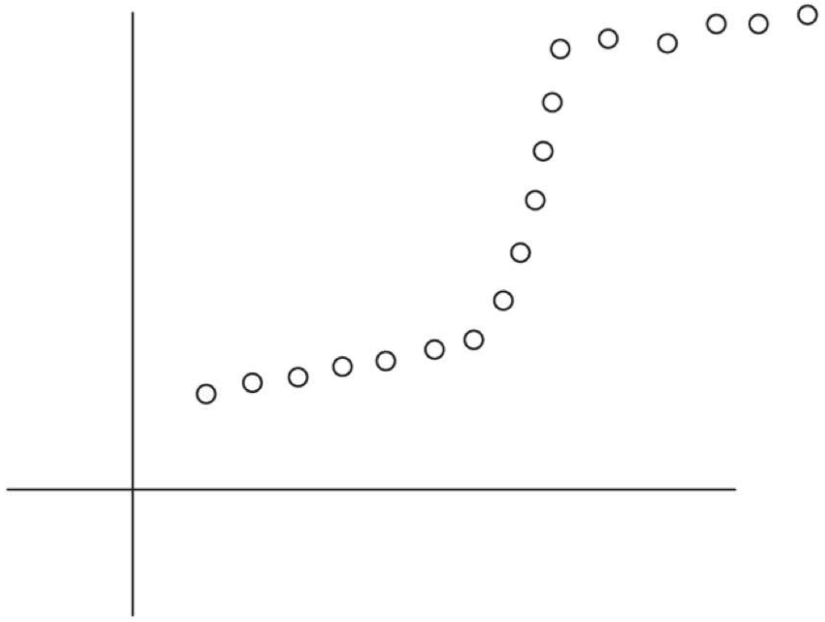


Figure 6.8 A set of points that lie approximately on three lines.

RNA Structure

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
▷ RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

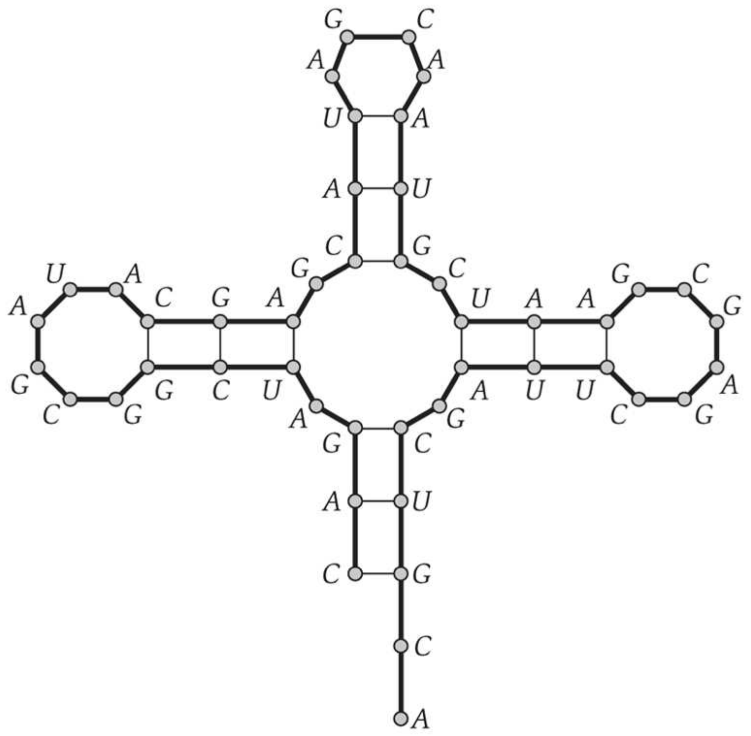


Figure 6.13 An RNA secondary structure. Thick lines connect adjacent elements of the sequence; thin lines indicate pairs of elements that are matched.

RNA Structure

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
▷ RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

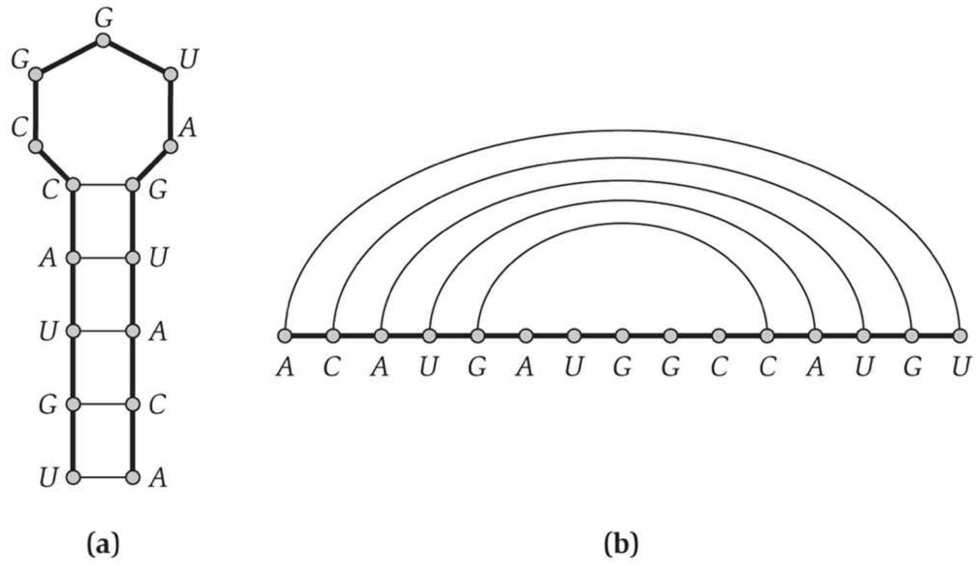


Figure 6.14 Two views of an RNA secondary structure. In the second view, (b), the string has been "stretched" lengthwise, and edges connecting matched pairs appear as noncrossing "bubbles" over the string.

Try this: ACCGGUAGU

Edit Distance

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
▷ Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

- Given two strings “lqorihten” and “algorithm”, can you tell how similar these strings are?
- Edit distance is the number of operations (deletions, insertions, substitutions) that you can convert from one string to the other.

```
S – N O W Y          – S N O W – Y
S U N N – Y          S U N – – N Y
  Cost: 3              Cost: 5
```

- How do you compute the smallest edit distance between two strings?
  - Brute force method? What’s the time complexity?
  - Greedy algorithm?
  - Dynamic programming

Edit Distance

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
▷ Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

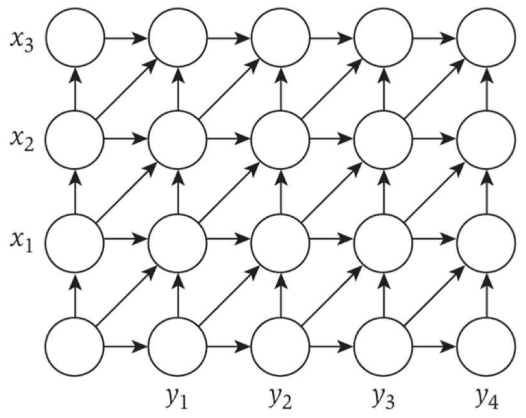


Figure 6.17 A graph-based picture of sequence alignment.

Edit Distance

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
▷ Edit Distance
Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

- □ example: mean vs. name
- □ $\delta = 2$, matching cost: 0 if same character, 1 if vowel to vowel, consonant to consonant, 3 if vowel to consonant.

Edit Distance

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
▷ Edit Distance
Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

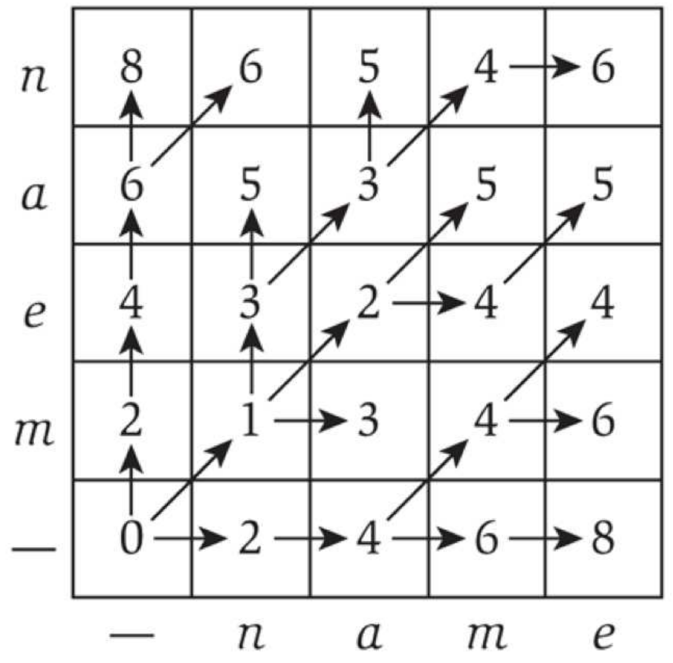


Figure 6.18 The OPT values for the problem of aligning the words *mean* to *name*.

Edit Distance

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
▷ Edit Distance
Edit Distance
Bellman-Ford Shortest Path Algorithm

- **Observation**: Given two strings $x[1 \cdots n]$ and $y[1 \cdots m]$. No matter how we match x to y, at the end of the match, we can only have:
  –
  –
  –
  – **Question**: Is it possible $x[n]$ matches to $y[i < m]$? or $y[m]$ matches to $x[j < n]$?
- **Example**: EXPONENTIAL vs. POLYNOMIAL
  – What are possible endings?
  – What are the subproblems we should consider?
  – How do we get an optimal answer?

Edit Distance

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
▷ Edit Distance
Bellman-Ford Shortest Path Algorithm

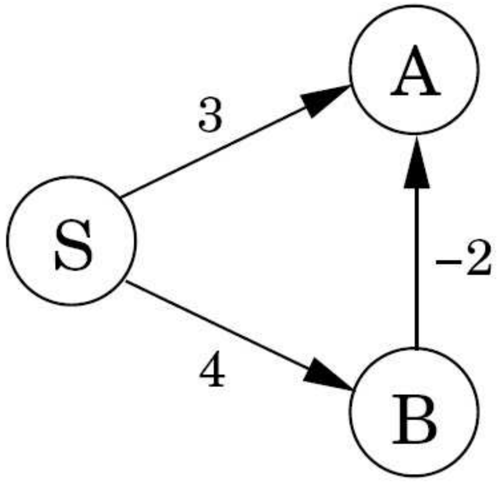
- Let $E(i,j)$ be the edit distance for the subproblem of strings with lengths i and j
- $E(i,j) = \min\{E(i-1,j-1)+\text{diff}(x[i],y[j]), E(i-1,j)+1, E(i,j-1)+1\}$

	□	P	O	L	Y	N	O	M	I	A	L
□											
E											
X											
P											
O											
N											
E											
N											
T											
I											
A											
L											

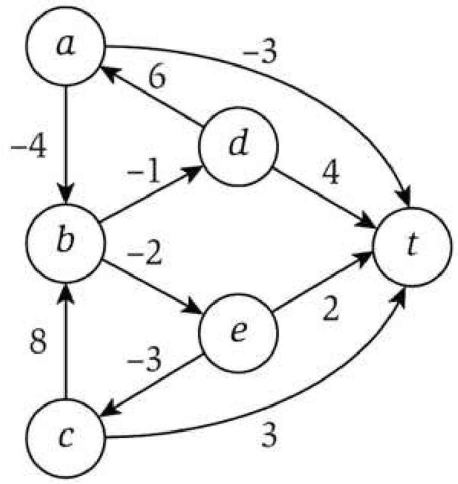
Bellman-Ford Shortest Path Algorithm

Dynamic Programming
Optimization
Weighted Interval Scheduling
Weighted Interval Scheduling
Knapsack Problem
Knapsack Problem
Knapsack Problem and Memory Functions
Piecewise Linear Regression
Piecewise Linear Regression
RNA Structure
RNA Structure
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
Edit Distance
▷ Bellman-Ford Shortest Path Algorithm

Can we simply add 2 to all edges so there will be no positive edges?



Try bellman-ford on this example:


a
b
c
d
e
t
−3
6
−4
−1
4
−2
2
8
−3
3