Ch1. SEQUENCE ALIGNMENT ALGORITHMS

9/30/2025

1.1. Global Alignment Algorithms
1.2. Heuristic interpretation of alignment score "likelihood"
1.3. Scoring schemes = math models of evolution
BLOSUM, PAM
1.4. Local Alignment Algorithms
1.5. Gaps in Alignment & Affine Gap Alignment algorithms
1.6. Shortest paths in graphs: Dijkstra's Algorithm
Topological sorting and optimal/linear-time shortest path algorithms

* pairwise sequence alignment

Pillars: CS, BIO, STAT

Dijkstra's shortest path algorithm

$G = \langle N, A \rangle$ directed graph

N: set of nodes/vertices
A: set of edges

Each edge has a non-negative length (cost)
one node: source node

Problem:

GIVEN:
$$\begin{cases} G = \langle N, A \rangle \quad N = \{1, 2, \dots, n\} \\ L[i,j] = \text{cost of edge } (i,j) \in A; \\ L[i,j] \geq 0 \quad \forall i,j \\ 1 = \text{source node} \end{cases}$$

FIND/COMPUTE: the cost/length of the shortest paths from the source to each of the other nodes in the graph and the shortest paths

The Algorithm: two sets of nodes: C, S
S = set of nodes already chosen
C = set of candidate nodes

at any step in the alg.:
→ S contains all the nodes whose shortest/minimum cost from the source is already known
→ C = rest of the nodes
at every step we choose a node in C whose cost to the source is smallest and add to S

Def: a path from the source to a node v is called special path if all intermediary nodes along the path are in S.

D matrix contains the length of the shortest special path for each node in the graph when we add to S a new node v from C, then the shortest special path to v is also the shortest path in the graph

Dijkstra's shortest path algorithm

Initialization: 1 is the source node

$S = \{1\}$

$C = \{2, 3, \dots, n\}$

D: cost of paths, in the end opt costs

$N = \{1, 2, \dots, n\}$

$L[i,j] \geq 0$ cost of edge $(i,j) \in A$

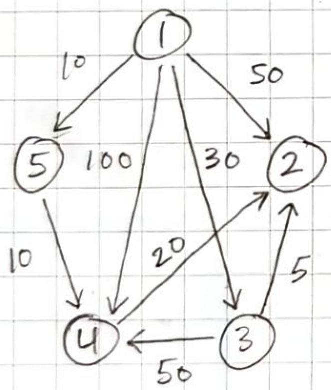
$L[i,j] = \infty$ if edge $(i,j) \notin A$

$P[i] = 1, \quad i = 2, 3, \dots, n$

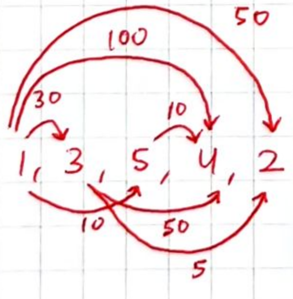
$P[2 \dots n]$

FOR $i = 2$ **TO** n **DO**
    $D[i] = L[1, i]$

{ GREEDY LOOP }
REPEAT $n-2$ times
    v = node in C with min $D[v]$
    $S = S \cup \{v\}$
    $C = C - \{v\}$
    FOR each $w \in C$ **DO**
        IF $D[w] > D[v] + L[v,w]$
            THEN $D[w] = D[v] + L[v,w]$
            $P[w] = v$



Step	v	C	D
initialization	–	{2,3,4,5}	[50, 30, 100, 10] (2 3 4 5)
1	5	{2,3,4}	[50, 30, 20, 10]
2	4	{2,3}	[40, 30, 20, 10]
3	3	{2}	[35, 30, 20, 10]



DAG Topological sorting