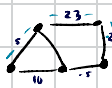


8 October

topics of computational biology

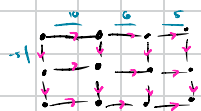
CS Bio Prob & stats

longest paths problem & shortest path problems



Cost: $5 + 23 + 2 = 26$

→ Undirected path



Directed path graph: Cost: 10

longest path: path of maximum cost ie max score alignment

Shortest path problem: minimize cost

→ Solvable in polynomial time $O(V^2)$

→ Dijkstra's algorithm

for DAG's (directed acyclic graphs):

• $O(|V| + |E|)$
linear time

• Can solve optimal shortest + longest path with same algorithm

→ by just multiplying all costs by -1

Longest path problem is NP-complete, in general (not DAG)

• Completely intractable: cannot be solved except for brute force

• Famous problem is P vs NP; $P \neq NP$ is what we believe

Shortest Path Algo - Dijkstra's Algorithm



→ find shortest path to every node

$G = (V, E)$ directed graph

V : set of vertices

E : set of edges

→ each edge has a non-negative cost/length

→ one node is source vertex

Problem: given (V, E) , $V = \{1, 2, \dots, n\}$, E edges (i, j) ,

Cost of edge $(i, j) = C(i, j)$

Compute: the cost of the shortest paths from source vertex to each of the other nodes in G , and the shortest paths themselves

Algorithm: two sets S, C of vertices

S : set already chosen

C : set of candidate vertices

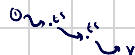
S : contains all the vertices whose minimal path cost from the source is already known

C : Rest of vertices

At each step, we choose a node in C , whose cost of the path to the source and add it to S

Def: A path from the source node to a node V is

called special if all intermediate nodes along path are in S



When we take a new node from C , V , to S - then the

shortest special path to V is the shortest path in G

The algorithm ends with all nodes in S

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

$$L(i, j) \geq 0 \text{ if edge } (i, j) \text{ exists in } G$$

Cost of edge (i, j)

$$L(i, j) = \infty \text{ if } (i, j) \notin E$$

D gives us solution: Shortest path Cost

{Initialization}

$$S = \{1\}$$

$$C = \{2, 3, \dots, n\} \mid 1 = \text{source node}$$

For $i = 2$ to n do:

$$D[i] = L[1, i] \rightarrow \text{direct connection to source}$$

{Greedy Loop}

Repeat $n-2$ times

V = node in C with minimum

Value in $D[V]$

$\rightarrow$ shortest path
from source

$$S = S \cup \{V\}$$

$$C = C \setminus \{V\}$$

For $w \in C$ do:

$$\text{If } D[w] > D[V] + L[V, w]$$

then

$$D[w] = D[V] + L[V, w]$$