Data Structures and Algorithms

Lecture 32

Aniket Basu Roy

2026-04-15 Wed

Contents

1 Agenda

1.1 Bellman-Ford Algorithm

- Motivation: handling negative edge weights
- Algorithm and time complexity
- Proof of correctness
- Detecting negative-weight cycles

2 Motivation

Dijkstra's requires all edge weights to be non-negative. Bellman-Ford handles **arbitrary** edge weights — including negative weights.

2.1 Negative-Weight Cycles

A **negative-weight cycle** reachable from s makes shortest paths ill-defined: traversing the cycle repeatedly gives paths of arbitrarily negative weight.

Bellman-Ford detects negative-weight cycles and reports them. If none are reachable from s, it computes $\delta(s, v)$ for all v.

3 Bellman-Ford Algorithm

Observation: If no negative-weight cycle is reachable from s, every shortest path is simple (no repeated vertices), hence uses at most $|V| - 1$ edges.

Idea: Relax every edge $|V| - 1$ times. After k rounds of relaxing all edges, all shortest paths using $\leq k$ edges have been correctly computed.

```
BELLMAN-FORD(G, w, s)
INITIALIZE-SINGLE-SOURCE(G, s)
for i = 1 to |G.V| - 1
    for each edge (u, v) in G.E
        RELAX(u, v, w)
for each edge (u, v) in G.E       // negative-cycle check
    if v.d > u.d + w(u, v)
        return FALSE              // negative-weight cycle reachable
return TRUE
```

Returns TRUE and correct $v.d = \delta(s, v)$ values if no negative-weight cycle is reachable; FALSE otherwise.

4 Time Complexity

Step	Cost
INITIALIZE-SINGLE-SOURCE	$O(V)$
$(n-1)$ passes over all E	$O(VE)$
Final check	$O(E)$

$$\boxed{T_{\text{Bellman-Ford}} = O(VE)}$$

Slower than Dijkstra's ($O(E \log V)$), but handles negative weights. For dense graphs ($E = \Theta(V^2)$): $O(V^3)$.

5 Proof of Correctness

5.1 Path-Relaxation Property

After i passes of the outer loop (lines 2–4), for every vertex v:

$$v.d \le \delta^{(i)}(s,\, v)$$

where $\delta^{(i)}(s, v)$ denotes the minimum weight of any path from s to v using at most i edges.

5.2 Proof by induction on i

Base case ($i = 0$): After INITIALIZE-SINGLE-SOURCE: $s.d = 0 = \delta^{(0)}(s, s)$, and $v.d = \infty = \delta^{(0)}(s, v)$ for $v \ne s$ (no 0-edge path exists).

Inductive step: Assume after $i-1$ passes, $v.d \le \delta^{(i-1)}(s, v)$ for all v.

Let $p = \langle s = v_0, v_1, \ldots, v_k \rangle$ $(k \le i)$ be an optimal path using $\le i$ edges. The prefix $s \rightsquigarrow v_{k-1}$ uses $\le i-1$ edges, so by induction:

$$v_{k-1}.d \le \delta^{(i-1)}(s, v_{k-1}) = w(p[s \to v_{k-1}]).$$

In pass i, we relax (v_{k-1}, v_k):

$$v_k.d \;\le\; v_{k-1}.d + w(v_{k-1}, v_k) \;\le\; w(p) \;=\; \delta^{(i)}(s, v_k). \quad \blacksquare$$

5.3 Corollary

After $|V|-1$ passes (assuming no negative-weight cycles reachable from s):

$$v.d = \delta(s, v) \quad \text{for all } v \in V.$$

Proof: All simple shortest paths use $\le |V|-1$ edges, so $\delta^{(|V|-1)}(s, v) = \delta(s, v)$. Combined with $v.d \ge \delta(s, v)$ (upper-bound property): equality holds. $\blacksquare$

6 Detecting Negative-Weight Cycles

6.1 Claim

After $|V| - 1$ passes, there exists an edge (u, v) with $v.d > u.d + w(u, v)$ if and only if a negative-weight cycle is reachable from s.

6.2 Proof ($\Rightarrow$)

If no negative-weight cycle is reachable: by correctness, $v.d = \delta(s, v)$ for all v. The triangle inequality gives $\delta(s, v) \le \delta(s, u) + w(u, v)$, so no edge can be further relaxed.

6.3 Proof ($\Leftarrow$)

Suppose there is a negative-weight cycle $C = \langle c_0, c_1, \ldots, c_k = c_0 \rangle$ reachable from s. Assume for contradiction that $c_i.d \le c_{i-1}.d + w(c_{i-1}, c_i)$ for all i. Sum around the cycle:

$$\sum_{i=1}^{k} c_i.d \;\le\; \sum_{i=1}^{k} c_{i-1}.d \;+\; w(C).$$

The left-hand sum equals the right-hand sum (same vertices), so $0 \le w(C) < 0$. Contradiction. ■

7 Example

Graph: $V = \{s, t, x, y, z\}$, edges from CLRS Figure 22.4:

```
s->t (6),  s->y (7),
t->x (5),  t->y (8),  t->z (-4),
x->t (-2),
y->x (-3), y->z (9),
z->s (2),  z->x (7)
```

After 4 passes ($|V| - 1 = 4$), the final distances are: $\delta(s, s) = 0$, $\delta(s, t) = 2$, $\delta(s, x) = 4$, $\delta(s, y) = 7$, $\delta(s, z) = -2$.

(The negative edge $t \to z$ creates a shortcut; $x \to t$ enables further improvements in later passes.)

8 Questions

- Trace Bellman-Ford on the graph from CLRS Figure 22.4, pass by pass.
- Why is it sufficient to run exactly $|V| - 1$ passes rather than more?
- Can Bellman-Ford detect a negative-weight cycle that is **not** reachable from s? Should it?