Bellman–Ford is optimal

for shortest hop-bounded paths

Tomasz Kociumaka **Adam Polak**

MAX PLANCK INSTITUTE FOR INFORMATICS

Bocconi

Bellman–Ford algorithm

$d[1 ... n] \leftarrow [+\infty, +\infty, \ldots, +\infty];$

$d[source] \leftarrow 0;$

for i **from** 1 **to** $n - 1$ **do**

  foreach edge $(u, v) \in E$ **do**

    $d[v] \leftarrow \min\{d[v], d[u] + w(u, v)\};$

Bellman–Ford algorithm

$d^{(0)}[1...n] \leftarrow [+\infty, +\infty, \ldots, +\infty]$;

$d^{(0)}[source] \leftarrow 0$;

for i **from** 1 **to** $n-1$ **do**

  foreach edge $(u, v) \in E$ **do**

    $d^{(i)}[v] \leftarrow \min\{d^{(i)}[v], d^{(i-1)}[u] + w(u, v)\}$;

Bellman–Ford algorithm

$d^{(0)}[1...n] \leftarrow [+\infty, +\infty, \ldots, +\infty]$;

$d^{(0)}[source] \leftarrow 0$;

for i **from** 1 **to** $n-1$ **do**

    $d^{(i)} \leftarrow d^{(i-1)}$;

    foreach edge $(u, v) \in E$ **do**

        $d^{(i)}[v] \leftarrow \min\{d^{(i)}[v], d^{(i-1)}[u] + w(u, v)\}$;

Bellman–Ford algorithm

$d^{(0)}[1...n] \leftarrow [+\infty, +\infty, \ldots, +\infty]$;

$d^{(0)}[source] \leftarrow 0$;

for i **from** 1 **to** $n-1$ **do**

  $d^{(i)} \leftarrow d^{(i-1)}$;

  foreach edge $(u, v) \in E$ **do**

    $d^{(i)}[v] \leftarrow \min\{d^{(i)}[v], d^{(i-1)}[u] + w(u, v)\}$;

$d^{(h)}[v] =$ shortest path from source to v using at most h hops

Bellman–Ford algorithm

$d^{(0)}[1...n] \leftarrow [+\infty, +\infty, \ldots, +\infty]$;

$d^{(0)}[source] \leftarrow 0$;

for i **from** 1 **to** ~~$n-1$~~ h **do**

  $d^{(i)} \leftarrow d^{(i-1)}$;

  foreach edge $(u, v) \in E$ **do**

    $d^{(i)}[v] \leftarrow \min\{d^{(i)}[v], d^{(i-1)}[u] + w(u, v)\}$;

$d^{(h)}[v]$ = shortest path from source to v using at most h hops

Shortest paths algorithms

Negative-weigth SSSP

$O(nm)$ [Ford ’56, Bellmann ’58]

Shortest paths algorithms

Negative-weigth SSSP

$O(nm)$ [Ford ’56, Bellmann ’58]

$O(\sqrt{n}m \log W)$ [Gabow ’85, Gabow–Tarjan ’89, Goldberg ’95]

$W =$ maximum edge weight

Shortest paths algorithms

Negative-weigth SSSP

$O(nm)$ [Ford '56, Bellmann '58]

$O(\sqrt{n}m \log W)$ [Gabow '85, Gabow–Tarjan '89, Goldberg '95]

$O(m^{1+o(1)} \log W)$ [Chen et al. '22]

$W =$ maximum edge weight

Shortest paths algorithms

Negative-weigth SSSP

$O(nm)$ [Ford ’56, Bellmann ’58]

$O(\sqrt{n}m \log W)$ [Gabow ’85, Gabow–Tarjan ’89, Goldberg ’95]

$O(m^{1+o(1)} \log W)$ [Chen et al. ’22]

$O(m \operatorname{polylog} m \log W)$ [Bernstein et al. ’22, Bringmann et al. ’23]

$W =$ maximum edge weight

Shortest paths algorithms

Negative-weigth SSSP	Shortest hop-bounded paths	
$O(nm)$	$O(hm)$	[Ford '56, Bellmann '58]
$O(\sqrt{n}m \log W)$		[Gabow '85, Gabow–Tarjan '89, Goldberg '95]
$O(m^{1+o(1)} \log W)$		[Chen et al. '22]
$O(m \operatorname{polylog} m \log W)$		[Bernstein et al. '22, Bringmann et al. '23]

W = maximum edge weight

Shortest paths algorithms

Negative-weigth SSSP	Shortest hop-bounded paths	
$O(nm)$	$O(hm)$	[Ford ’56, Bellmann ’58]
$O(\sqrt{n}m \log W)$	?	[Gabow ’85, Gabow–Tarjan ’89, Goldberg ’95]
$O(m^{1+o(1)} \log W)$		[Chen et al. ’22]
$O(m \text{ polylog } m \log W)$		[Bernstein et al. ’22, Bringmann et al. ’23]

$W =$ maximum edge weight

Our result

Unless the **APSP Hypothesis** fails,

no $O(h^{1-\varepsilon}m)$ or $O(hm^{1-\varepsilon})$ time algorithm

for **shortest** h**-hop-bounded** s-t **path**.

Our result

Unless the **APSP Hypothesis** fails,

no $O(h^{1-\varepsilon}m)$ or $O(hm^{1-\varepsilon})$ time algorithm

for **shortest** h**-hop-bounded** s-t **path**,

even in **undirected graphs** with **nonnegative edge weights**.

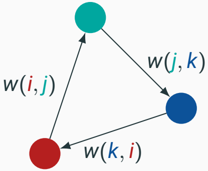
The negative triangle problem

Given (complete) graph on N nodes

with **edge weights w** $: [N] \times [N] \to \mathbb{Z}$

find $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k} \in [N]$

such that $\boldsymbol{w(i, j) + w(j, k) + w(k, i) < 0}$



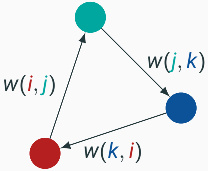
The negative triangle problem

Given (complete) graph on N nodes

with **edge weights** $\boldsymbol{w} : [N] \times [N] \to \mathbb{Z}$

find $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k} \in [N]$

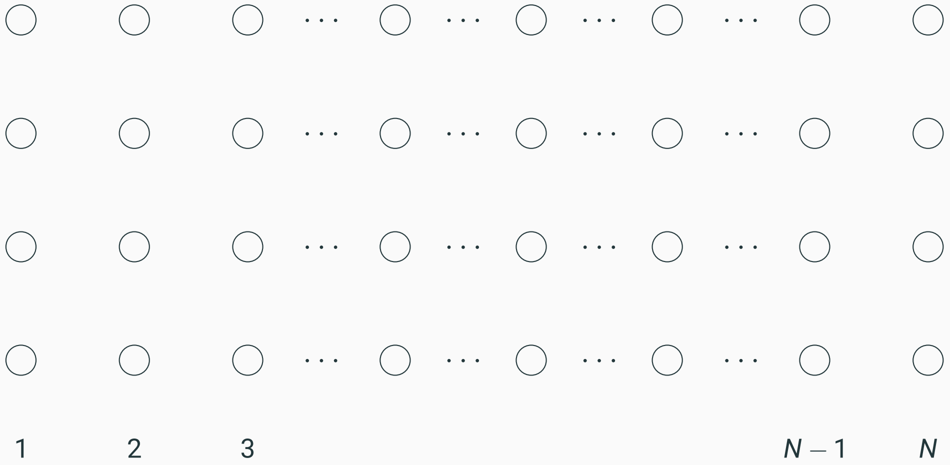
such that $\boldsymbol{w}(\boldsymbol{i},\boldsymbol{j}) + \boldsymbol{w}(\boldsymbol{j},\boldsymbol{k}) + \boldsymbol{w}(\boldsymbol{k},\boldsymbol{i}) < \mathbf{0}$



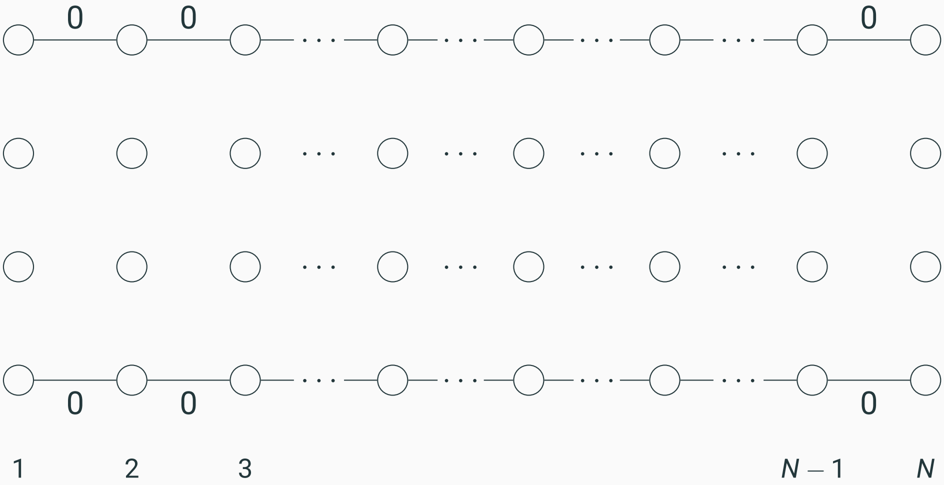
No $O(N^{3-\varepsilon})$ time algorithm unless the APSP hypothesis fails.

[Vassilevska Williams, Williams '10]

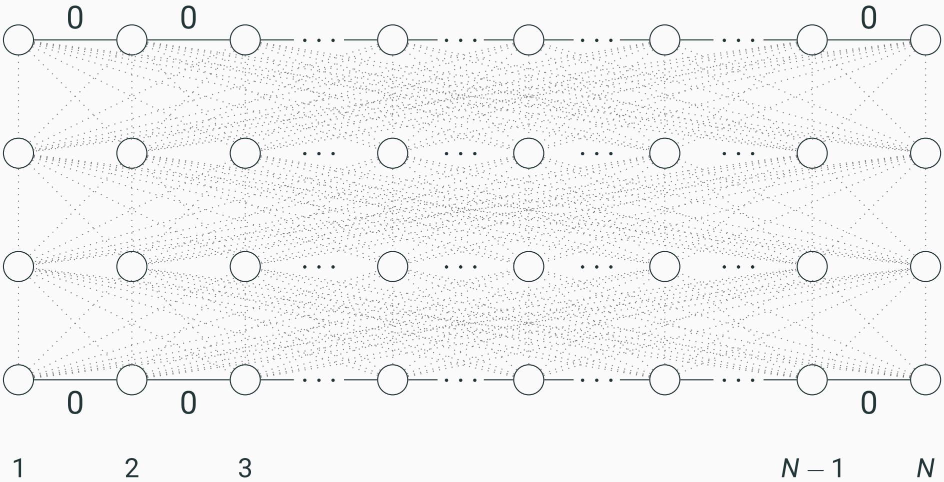
Reduction: negative triangle $\longrightarrow$ shortest hop-bounded path



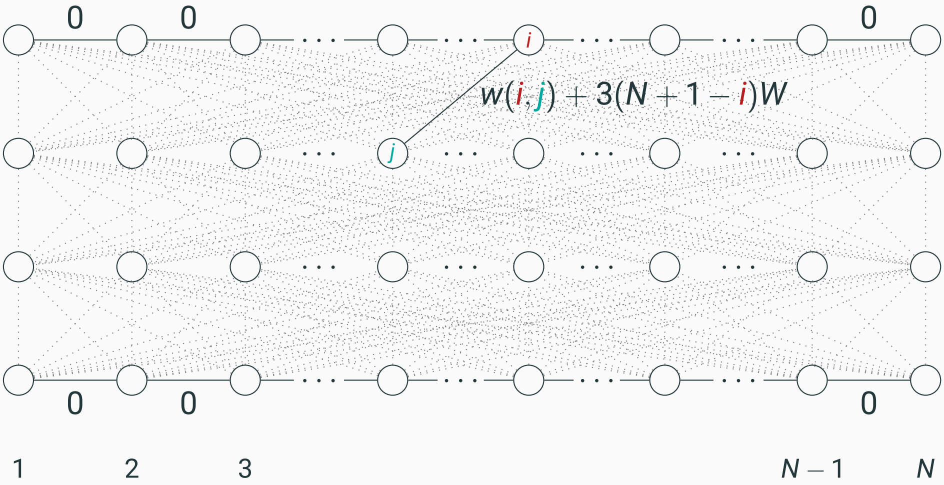
Reduction: negative triangle ⟶ shortest hop-bounded path



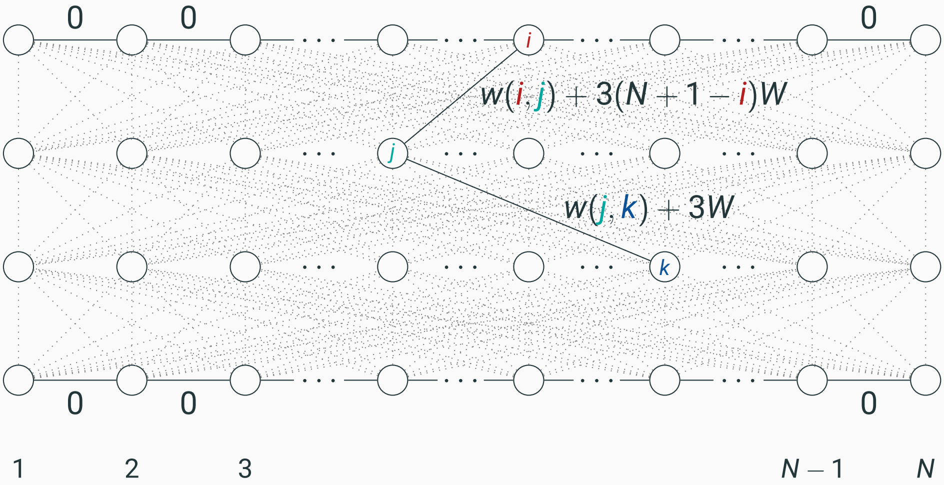
Reduction: negative triangle ⟶ shortest hop-bounded path



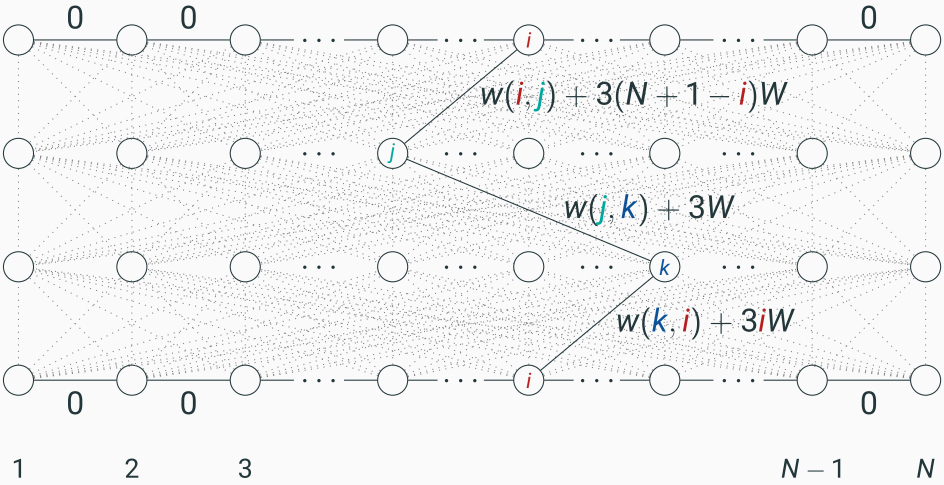
Reduction: negative triangle ⟶ shortest hop-bounded path



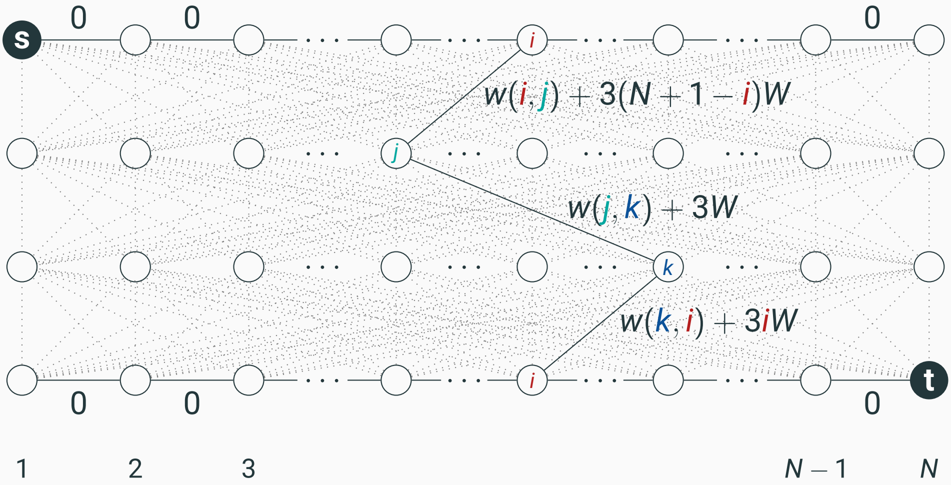
Reduction: negative triangle $\longrightarrow$ shortest hop-bounded path



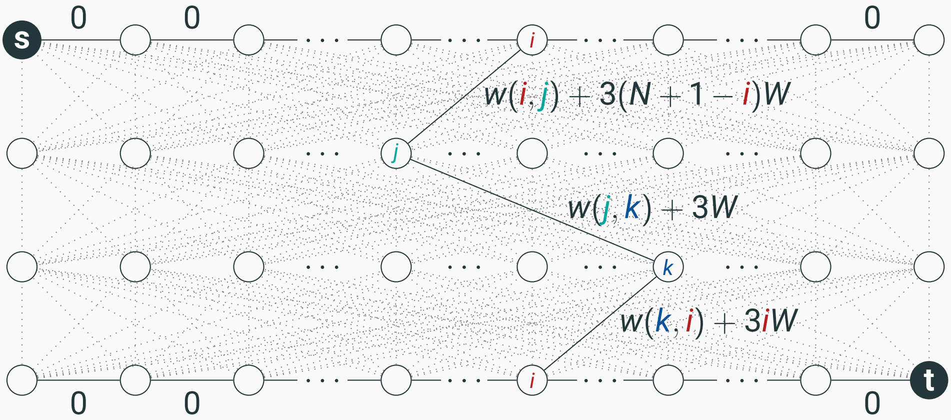
Reduction: negative triangle ⟶ shortest hop-bounded path



Reduction: negative triangle $\longrightarrow$ shortest hop-bounded path



Reduction: negative triangle $\longrightarrow$ shortest hop-bounded path



$\exists$ negative triangle $\iff \exists$ s-t path with $\leqslant N+2$ hops of length $< 3(N+2)W$

$\underbrace{\text{negative triangle}}_{i,j,k \,:\, w(i,j)+w(j,k)+w(k,i)<0}$

Reduction: negative triangle $\longrightarrow$ shortest hop-bounded path

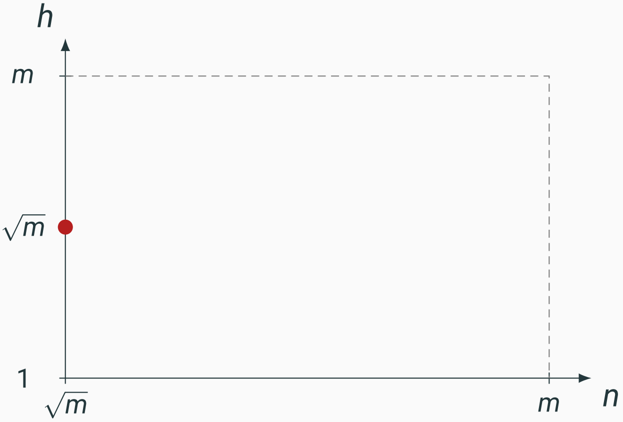
$$N \text{ nodes} \quad \longrightarrow \quad \begin{aligned} n &= O(N) \text{ nodes} \\ m &= O(N^2) \text{ edges} \\ h &= O(N) \text{ hop bound} \end{aligned}$$

Reduction: negative triangle $\longrightarrow$ shortest hop-bounded path

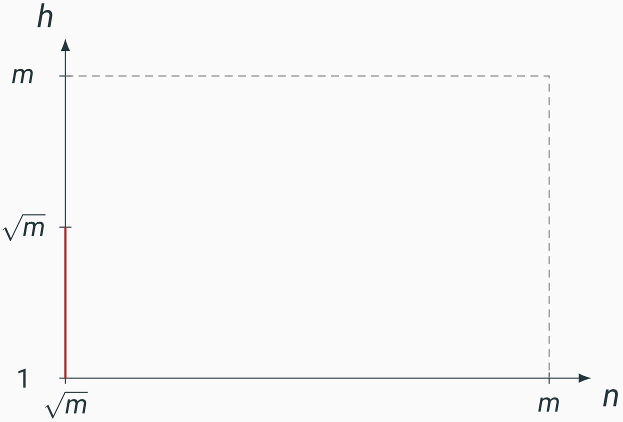
$$N \text{ nodes} \quad \longrightarrow \quad \begin{aligned} n &= O(N) \text{ nodes} \\ m &= O(N^2) \text{ edges} \\ h &= O(N) \text{ hop bound} \end{aligned}$$

$$O(N^{3-\varepsilon'}) \text{ time} \quad \longleftarrow \quad O(h^{1-\varepsilon} m) \text{ or } O(h m^{1-\varepsilon}) \text{ time}$$

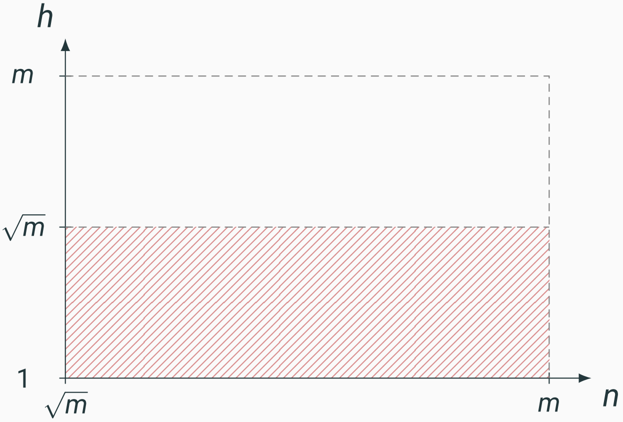
Hardness regime



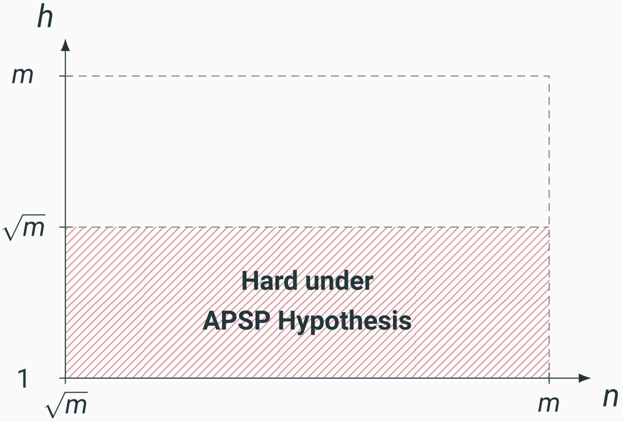
Hardness regime



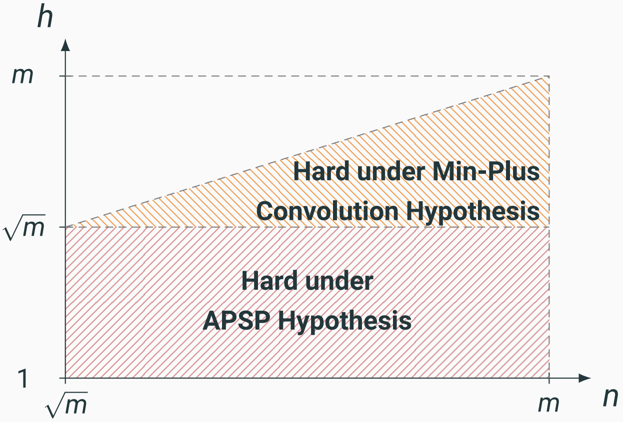
Hardness regime



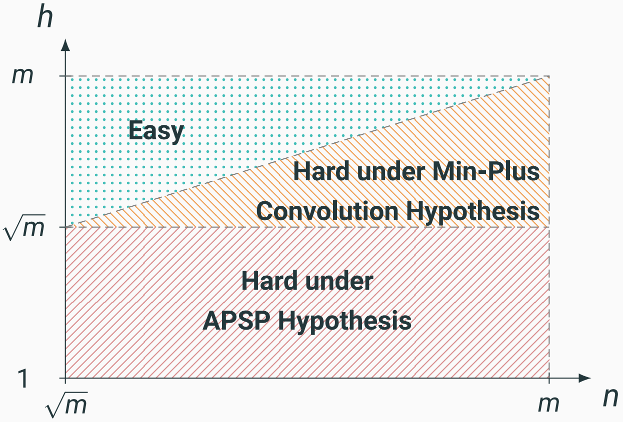
Hardness regime



Hardness regime



Hardness regime



Where else Bellman–Ford might be optimal?

Negative-weigth SSSP

$O(nm)$ [Ford ’56, Bellmann ’58]

$O(\sqrt{n}m \log W)$ [Gabow ’85, Gabow–Tarjan ’89, Goldberg ’95]

$O(m^{1+o(1)} \log W)$ [Chen et al. ’22]

$O(m \operatorname{polylog} m \log W)$ [Bernstein et al. ’22, Bringmann et al. ’23]

$W =$ maximum edge weight

Where else Bellman–Ford might be optimal?

Negative-weigth SSSP

$O(nm)$	[Ford '56, Bellmann '58]
$O(\sqrt{n}m \log W)$	[Gabow '85, Gabow–Tarjan '89, Goldberg '95]
$O(m^{1+o(1)} \log W)$	[Chen et al. '22]
$O(m \operatorname{polylog} m \log W)$	[Bernstein et al. '22, Bringmann et al. '23]

$W =$ maximum edge weight

Open: Negative-weight SSSP faster than $O(nm)$ without $\log W$ factor?

Thank you!