x

A Nearly Optimal Time-Space Lower Bound for Directed st-Connectivity on NNJAGs

Jeff Edmonds

ICSI Berkeley

Chung K. Poon

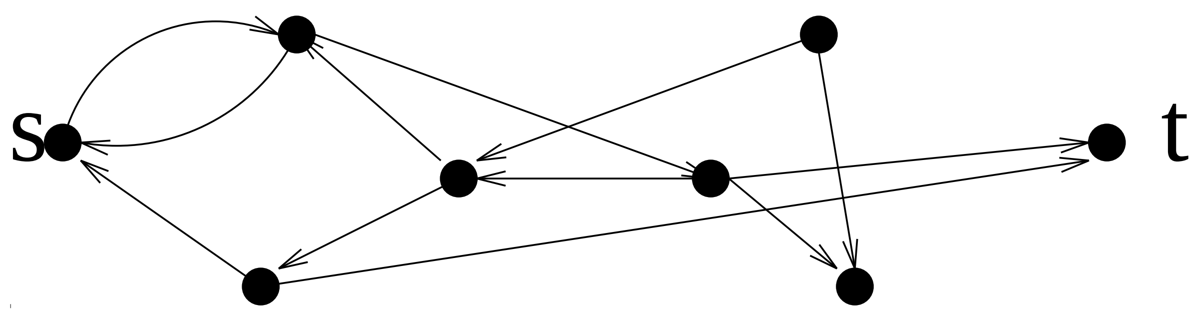
U. Toronto

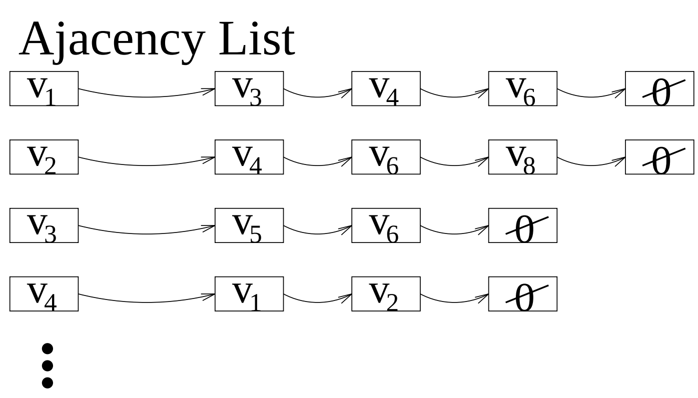
Task

Directed st-connectivity

Input: A directed graph
with distinguished nodes s and t

Output: "Yes" - $\exists$ a directed path from s to t
"No" - otherwise





Upper Bounds

Undirected

Depth First Search	Random Walk
Time $\leq O(m+n)$	Time $\leq O(mn)$
Space $\leq O(n \log n)$	Space $\leq O(\log n)$
(Tarjan ‘72)	(Aleliunas et al. ’79)

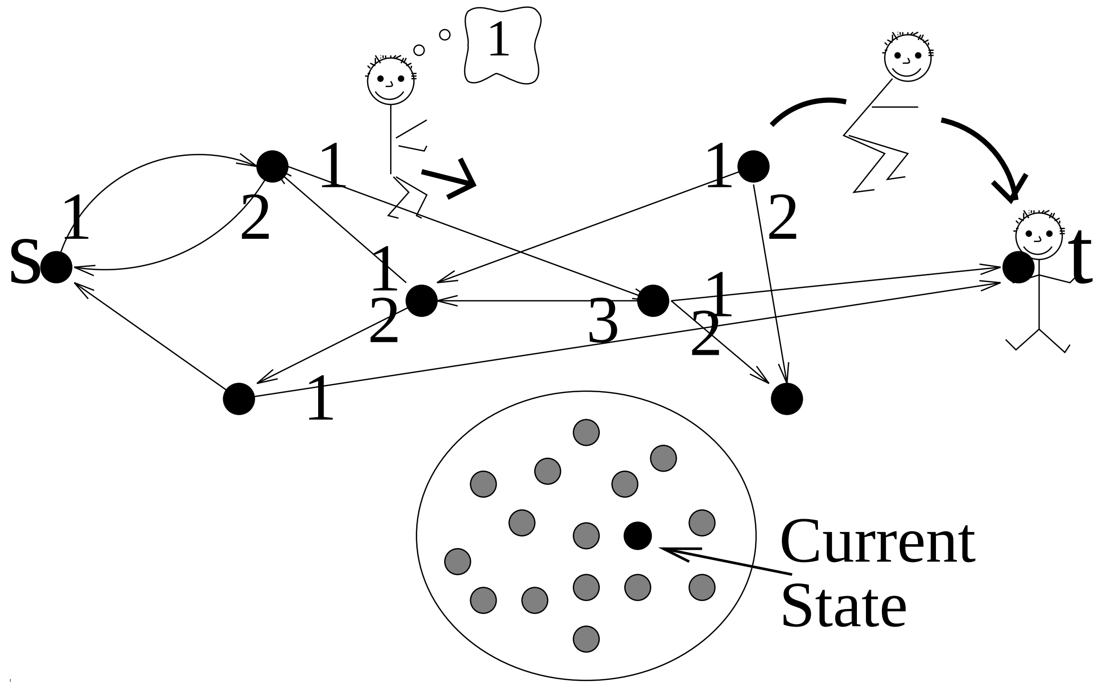
Time $\times$ Space $\leq O\left(mn \log^{O(1)} n\right)$ Broder et al. Barnes & Feige

Directed

Depth First Search	Combined	Recursive Doubling
Time $\leq O(m+n)$	Time $\leq n^{O(1)}$	Time $\leq n^{O(\log n)}$
Space $\leq O(n \log n)$	Space $\leq \frac{n}{2^{\Omega(\sqrt{\log n})}}$	Space $\leq O\left(\log^2 n\right)$
(Tarjan ‘72)	(Barnes et al. ’92)	(Savitch ’70)

Time $\in 2^{\Omega\left(\log^2\left(\frac{n}{S}\right)\right)} \times n^{O(1)}$ (Barnes et al. ’92)

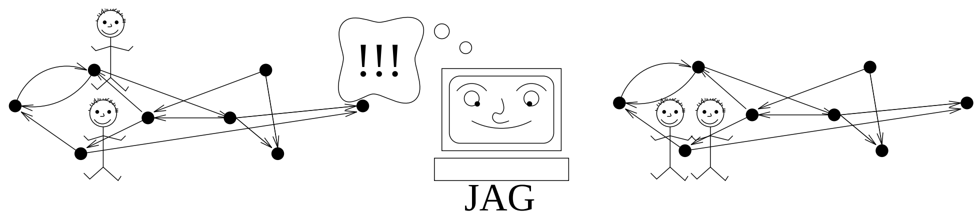
The JAG Model



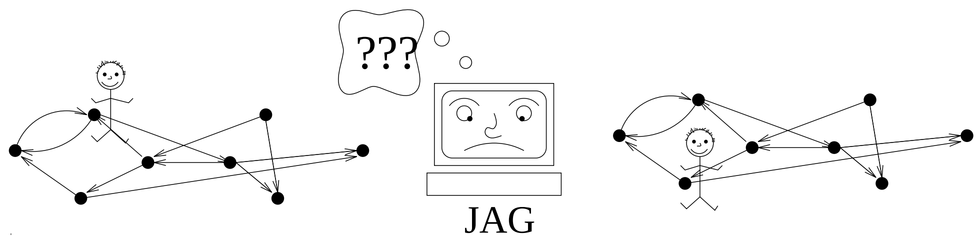
- Input includes labels $\in$ [1..out-deg.] on out-edges
- Pebbles placed on nodes of input graph
  - "walk" along directed edge specified by label
  - "jump" to the location of another pebble
- Non-uniform state change
- Space $= p \times \log(n) + \log(\# \text{ of states})$
  - p distinguishable pebbles
  - q states

The JAG Model

Partition of pebbles

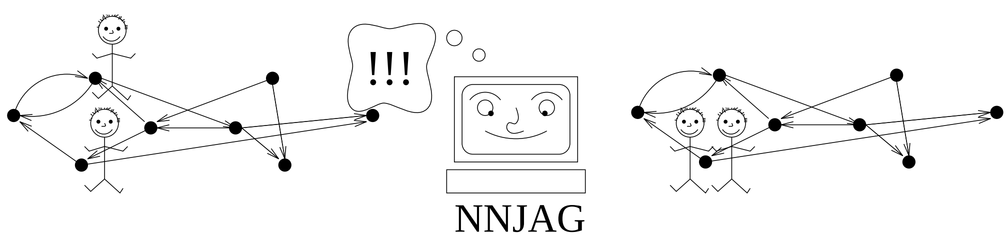


No node names

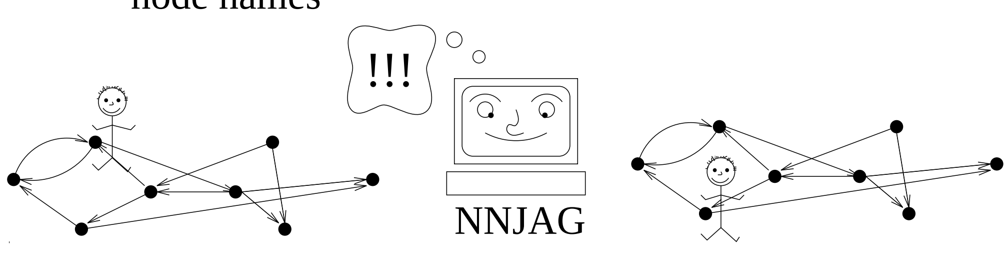


The Node-Named JAG Model

Partition of pebbles

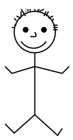


node names



Lower Bounds

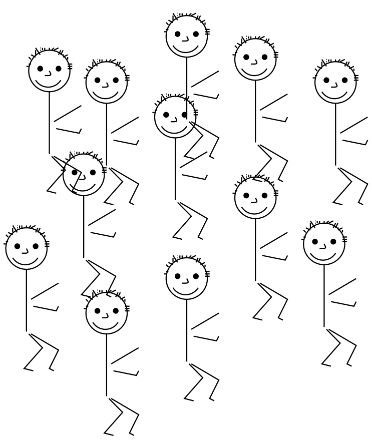
Undirected st-Connectivity



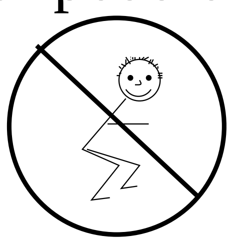
one pebble

$\text{Time} \geq \Omega\left(m^2\right)$

BRT '92



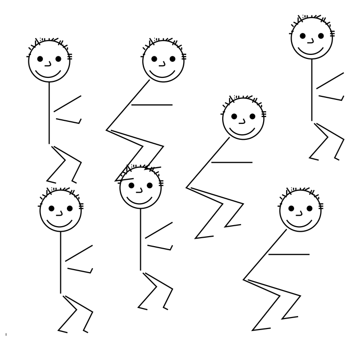
arbitrary number of pebbles



$\text{Degree} = 4p$

$\text{Time} \geq \Omega\left(m \log n\right)$

BBRRT '93



$\frac{\text{log n}}{\text{loglog n}}$ pebbles

$\text{Time} \geq \Omega\left(n^{1+\frac{1}{\log\log n}}\right)$

Edmonds '93

Lower Bounds

Directed st-Connectivity

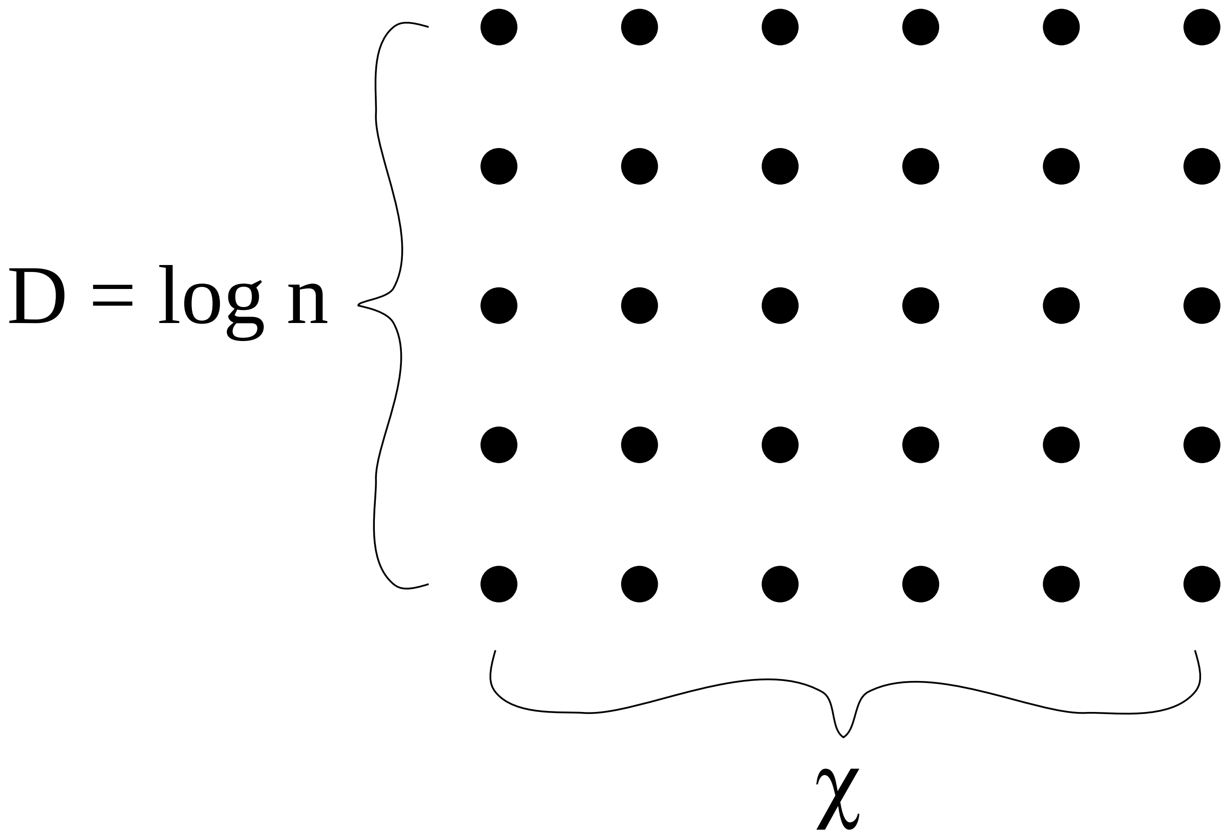
The JAG Model

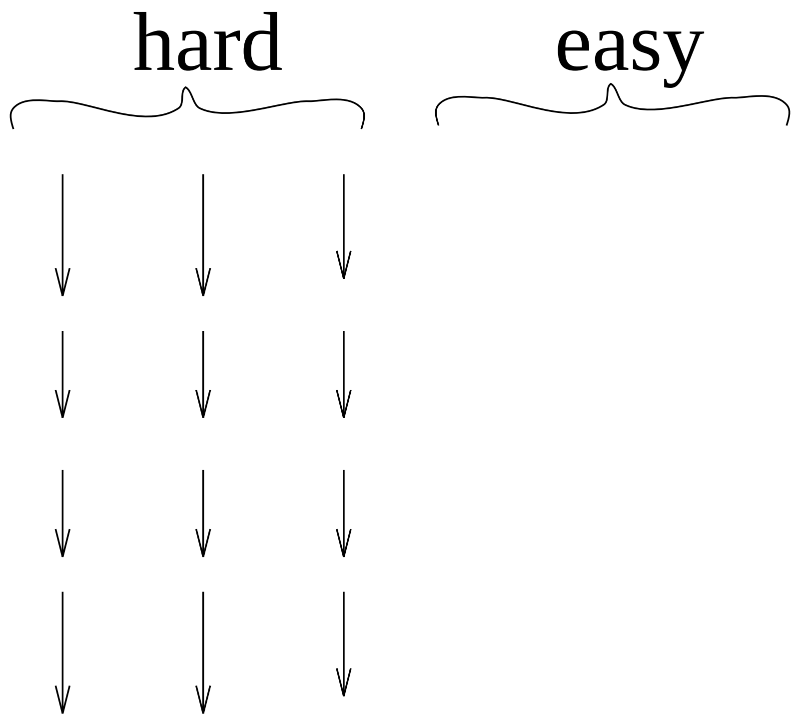
Space	$\geq \Omega\left(\frac{\log^2 n}{\log\log n}\right)$	Cook & Rackoff '80
Time $\times$ Space$^{\frac{1}{2}}$	$\geq \Omega\left(mn^{\frac{1}{2}}\right)$	Barnes & Edmonds '93
Time $\times$ Space	$\geq \Omega\left(\frac{n^2}{\log n}\right)$	

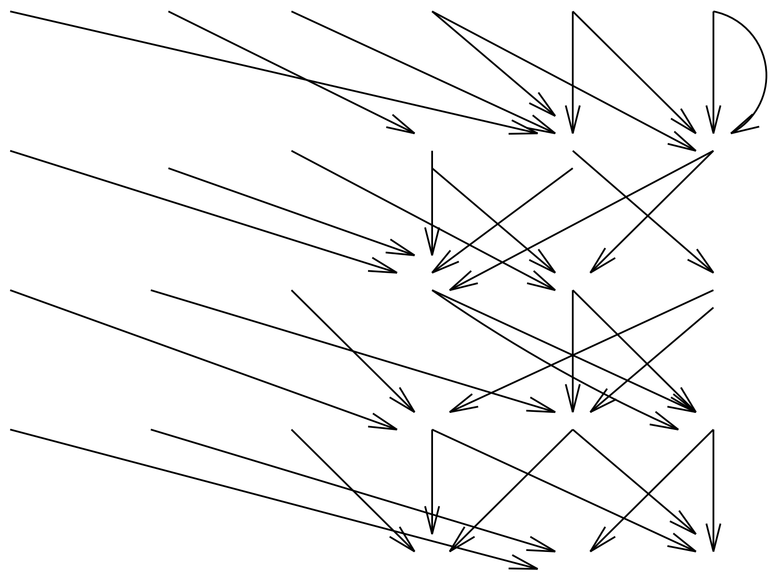
The Probabilistic Node-Named-JAG Model

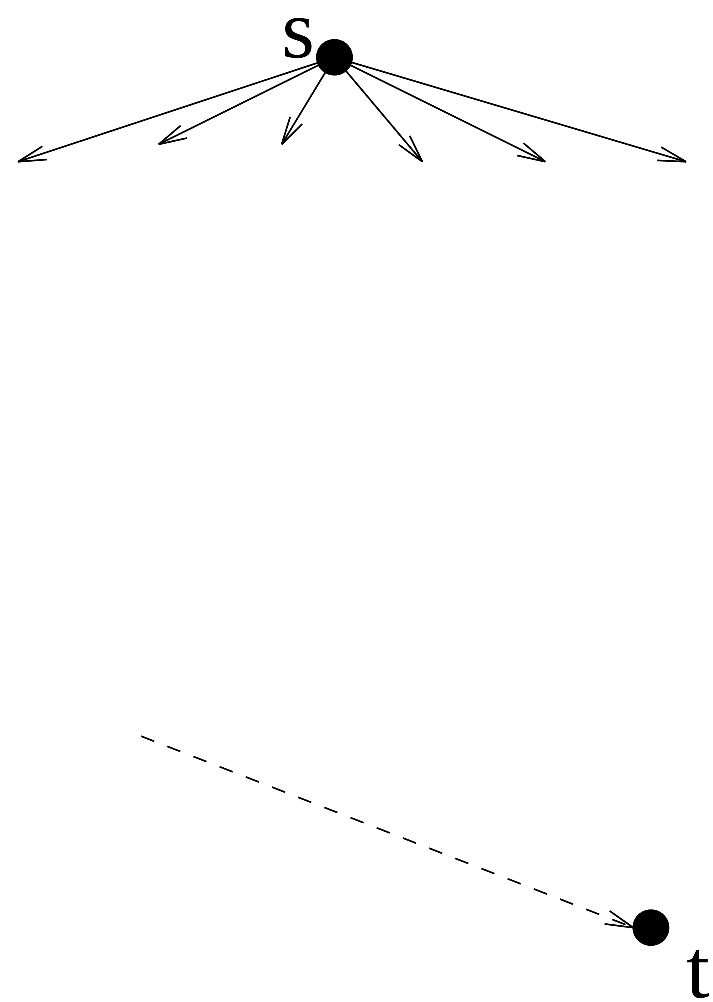
Space	$\geq \Omega\left(\frac{\log^2 n}{\log\log n}\right)$	Poon '93
Time $\times$ Space$^{\frac{1}{3}}$	$\geq \Omega\left(m^{\frac{2}{3}}n^{\frac{2}{3}}\right)$	Edmonds '93
Time	$\geq 2^{\Omega\left(\frac{\log^2\left(\frac{n}{S}\right)}{\log\log n}\right)}$	**Edmonds & Poon '95**

Layered Graphs with Hard Paths

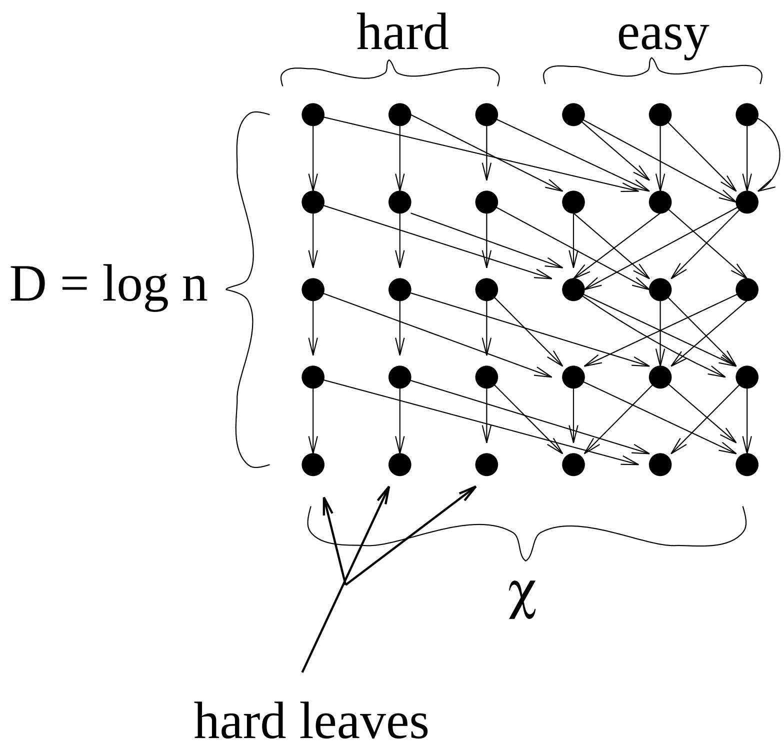



hard
easy




s
t

Progress

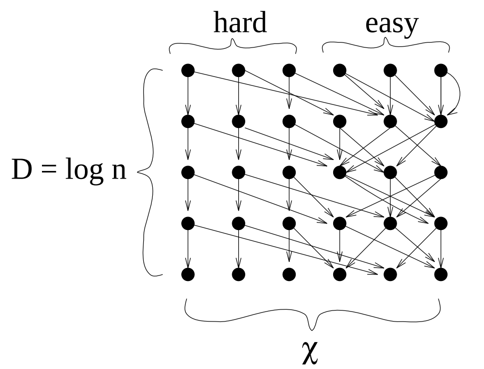


Def^n of Progress: Having a pebble on a hard leaf.

- Must "traverse" hard path to reach hard leaf
  (If initially no pebbles on hard path)

- Must make $\chi/2$ progress
  (else connect t to the hard node not reached.)

Steps of Proof



1. Time $= \chi/8$, Space $= \infty$, any start config $\langle Q, \Pi \rangle$,

   $\Pr_G\left[Prog \geq 3S\right] < \frac{1}{2}2^{-S}$

2. Time $= \frac{(\chi/8)\times(\chi/2)}{3S}$, Space $= S$,

   $\Pr_G\left[\ st\text{-conn}\ \right] < \frac{1}{2}$

   Conclude: $S \times T \in \Omega\left(\frac{n^2}{\log^2 n}\right)$.

3. Define Recursive Graph

4. Lower Bound for G_{k-1} (Time $= T_{k-1}$, Space $= S$)

   $\rightarrow$ Lower Bound for G_k (Time $= T_k$, Space $= S$)

   Conclude: $T \in 2^{\Omega\left(\frac{\log^2\left(\frac{n}{S}\right)}{\log\log n}\right)}$.

Step 2

Lemma:

Time $= \frac{(\chi/8)\times(\chi/2)}{3S}$, Space $= S$,

$\Pr_G[\ st\text{-conn}\ ] < \frac{1}{2}$

Conclude: $S \times T \in \Omega\left(\frac{n^2}{\log^2 n}\right)$.

Proof:

- Total progress needed is $\chi/2$

- Partition the computation into $\frac{(\chi/2)}{(3S)}$ periods.

- For each input G,
  there exists a period of $\chi/8$ time steps
  in which 3S progress is made.

- Given fixed $\langle Q, \Pi \rangle$ and $\chi/8$ time steps,
  $\Pr_G[Prog \geq 3S] < \frac{1}{2}2^{-S}$

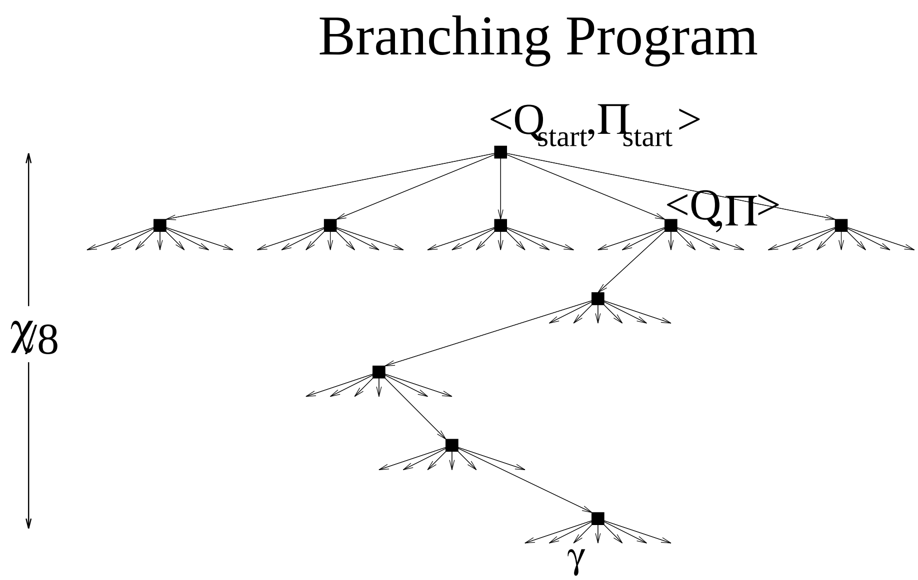
- $\Pr_G[\exists\, \langle Q, \Pi \rangle\,,\ s.t.\ Prog \geq 3S] < \frac{1}{2}2^{-S} \times 2^S = \frac{1}{2}$

An Attempt at Step 1

Lemma:

$$\text{Time} = \chi/8, \qquad \text{Space} = \infty, \qquad \text{any start config } \langle Q, \Pi \rangle,$$

$$\Pr_G [Prog \geq 3S] < \tfrac{1}{2} 2^{-S}$$



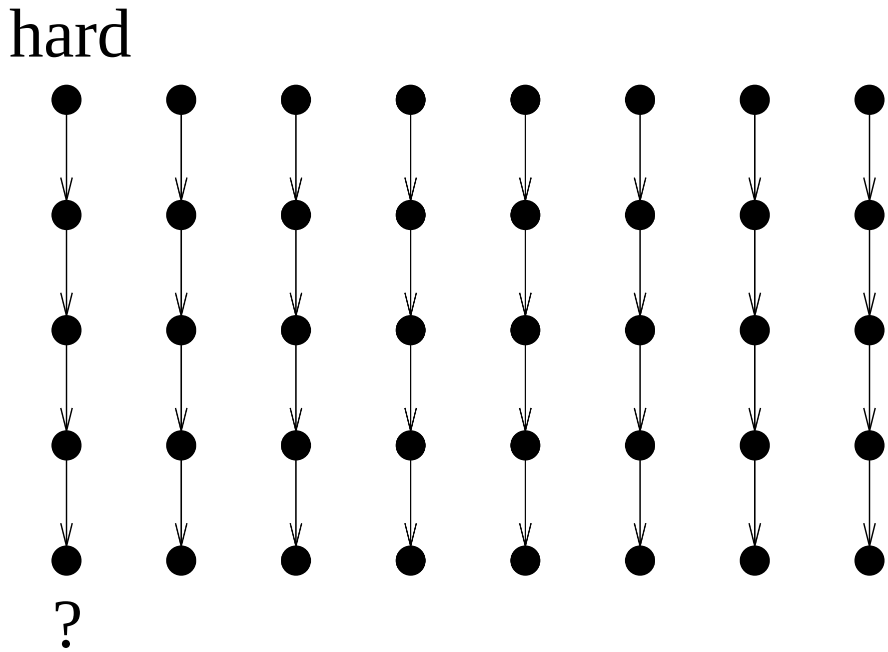
γ: Computation Path

$E(\gamma)$: Set of edges traversed by γ. $|E| \leq \chi/8$.

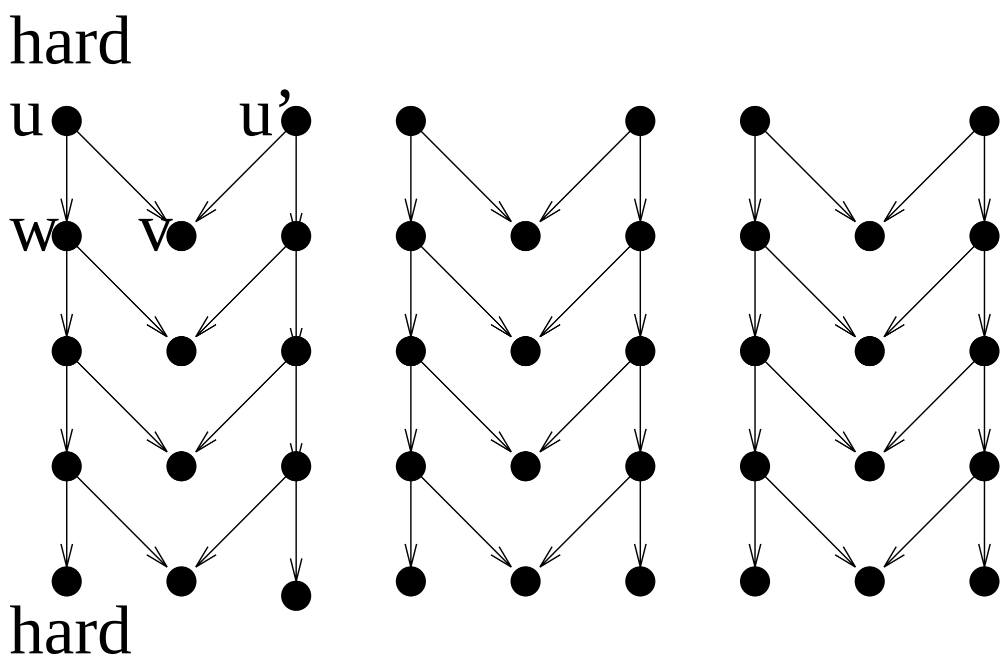
$$\begin{aligned} &\Pr_G [Prog \geq 3S] \\ &= \sum_\gamma \Pr_G [Prog \geq 3S \mid G \text{ follows } \gamma] \times \Pr_G [G \text{ follows } \gamma] \\ &\leq \max_\gamma \Pr_G [Prog \geq 3S \mid G \text{ follows } \gamma] \times 1 \end{aligned}$$

$$\max_{|E| \leq \chi/8} \Pr_G [Prog(G, E) \geq 3S \mid E \subseteq G]$$

A Good Set of Edges E



A Bad Set of Edges E

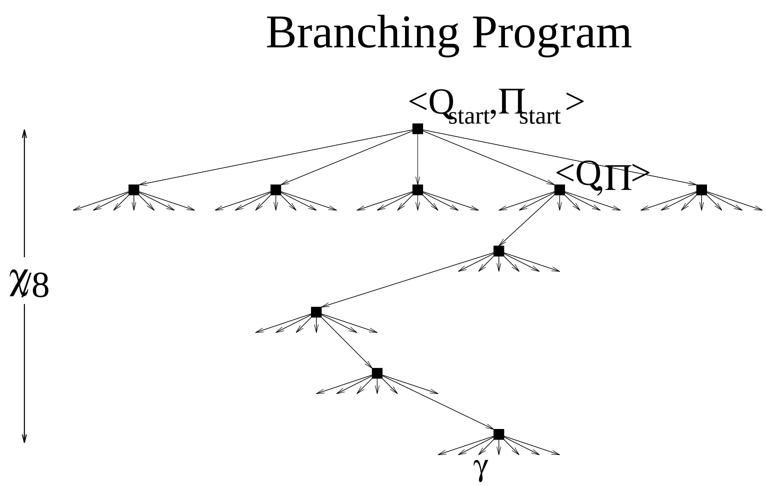


Step 1

Lemma:

Time $= \chi/8$, Space $= \infty$, any start config $\langle Q, \Pi \rangle$,

$$\Pr_G\left[Prog \geq 3S\right] < \tfrac{1}{2}2^{-S}$$



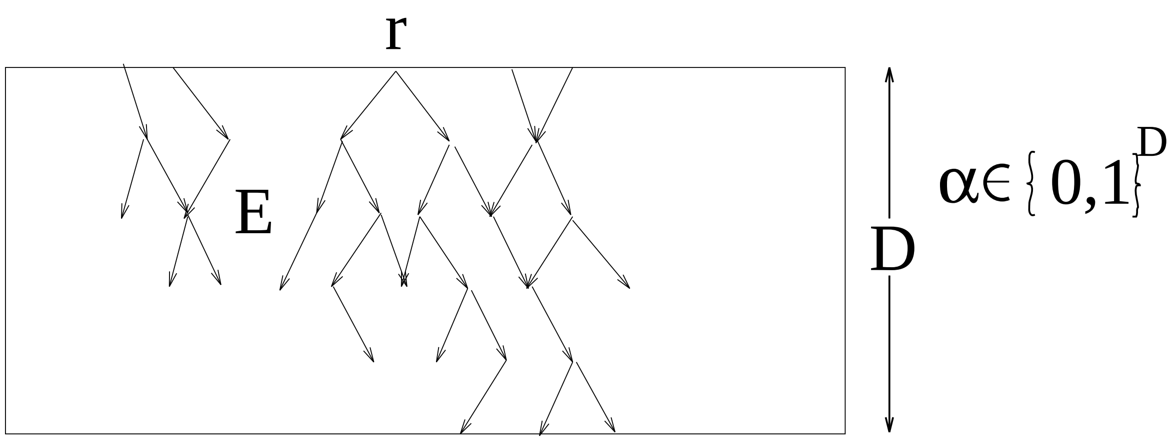
γ: Computation Path

$E(\gamma)$: Set of edges traversed by γ. $|E| \leq \chi/8$.

$$\begin{aligned}
&\Pr_G\left[Prog \geq 3S\right] \\
&= \sum_{\gamma \notin C} \Pr_G\left[Prog \geq 3S \mid G \text{ follows } \gamma\right] \times \Pr_G\left[G \text{ follows } \gamma\right] \\
&\quad + \sum_{\gamma \in C} \Pr_G\left[Prog \geq 3S \mid G \text{ follows } \gamma\right] \times \Pr_G\left[G \text{ follows } \gamma\right] \\
&\leq \max_{\gamma \notin C} \Pr_G\left[Prog \geq 3S \mid G \text{ follows } \gamma\right] \times 1 \\
&\quad + 1 \times \Pr_G\left[G \text{ follows } \gamma \in C\right] \\
&< \frac{1}{2}2^{-S}
\end{aligned}$$

Fix $|E| \le \chi/8$.

Bound $\Pr_G[Prog(G,E) \ge 3S \mid E \subseteq G]$



Classify $\alpha \in \{0,1\}^D$

$W_r \subseteq \{0,1\}^D$**:** cannot be hard path

$Y_r \subseteq \{0,1\}^D$**:** could be hard path and is traversed

$Z_r \subseteq \{0,1\}^D$**:** could be hard path, but not traversed

$$\Pr_G[Prog\ on\ r \mid E \subseteq G] \le \frac{|Y_r|}{|Y_r|+|Z_r|} \le \frac{\chi/8}{|Y_r|+|Z_r|}$$

Define Set of Computation Paths C

$E \in C$ if $|\{r \mid |Y_r| + |Z_r| \le 2^{0.1D}\}| \ge S$

$\underline{\text{Fix } |E| \leq \chi/8.}$

$\underline{\text{Bound } \Pr_G[Prog(G,E) \geq 3S \mid E \subseteq G]}$

$\Pr_G[Prog\ on\ r \mid E \subseteq G] \leq \frac{|Y_r|}{|Y_r|+|Z_r|} \leq \frac{\chi/8}{|Y_r|+|Z_r|}$

$E \in C$ if $|\{r \mid |Y_r| + |Z_r| \leq 2^{0.1D}\}| \geq S$

Three types of progress:

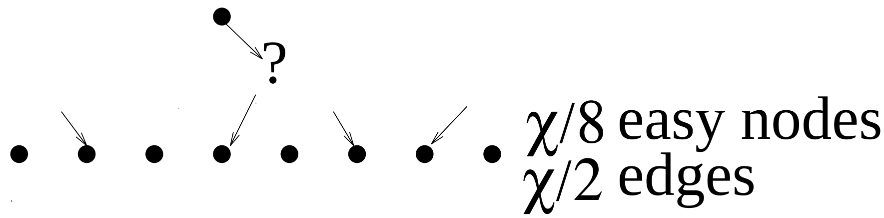
- free progress when pebble is initially on hard path
  ($\leq p \leq S$)
- high collision progress when $|Y_r| + |Z_r| \leq 2^{0.1D}$
  ($\leq S$, when $E \notin C$)
- low collision progress when $|Y_r| + |Z_r| > 2^{0.1D}$
  ($\Pr = \frac{\chi/8}{2^{0.1D}}$)

Bounding the Probability

If $E \notin C$,

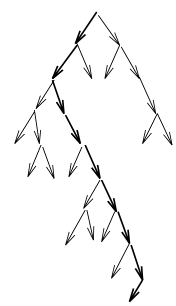
$\Pr_G[Prog(G,E) \geq 3S \mid E \subseteq G] < 2^{-\Omega(SD)}$

Probability of Single Collision



$\Pr_G[\text{ edge collides }] \leq \frac{\chi/8}{\chi/2} = \frac{1}{4}$.

$\Pr_G[G \text{ follows } \gamma \in C]$



Branching Processes

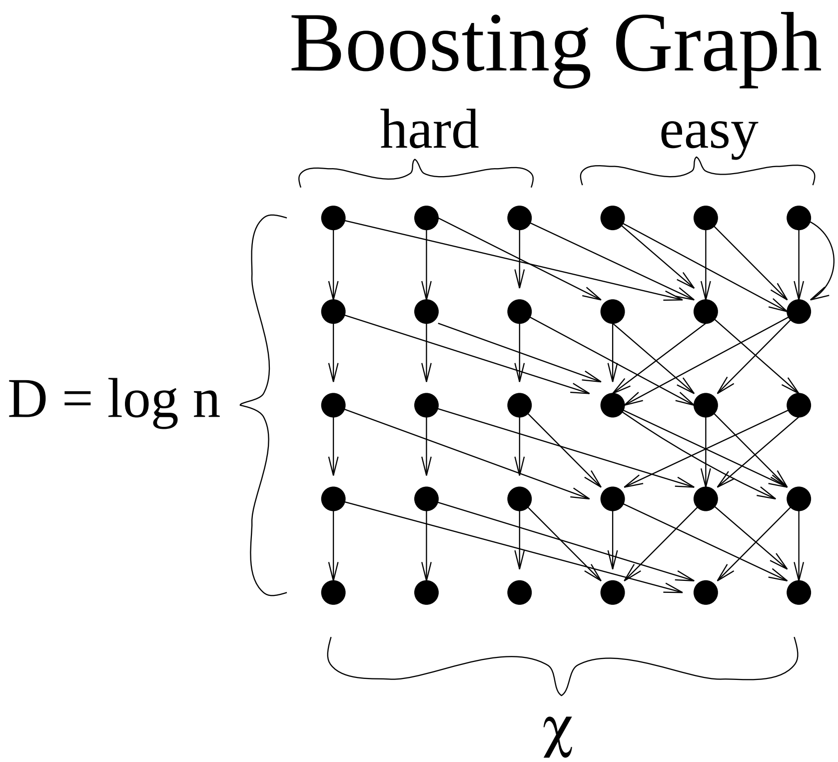
- Binary Tree
- One designated path that is guaranteed to live.
- Every other edge dies with prob $1/4$.
- Let "$\langle |Y| + |Z| \rangle$" be the # of leaves a level D.
- $\exp[\langle |Y| + |Z| \rangle] = (3/4 \times 2)^D = 2^{0.58D}$.
- $\Pr\left[\langle |Y| + |Z| \rangle \leq 2^{0.1D}\right] < 2^{-\Omega(D)}$.
- $\Pr\left[|\{r \mid \langle |Y| + |Z| \rangle_r \leq 2^{0.1D}\}| \geq S\right] < 2^{-\Omega(DS)}$.

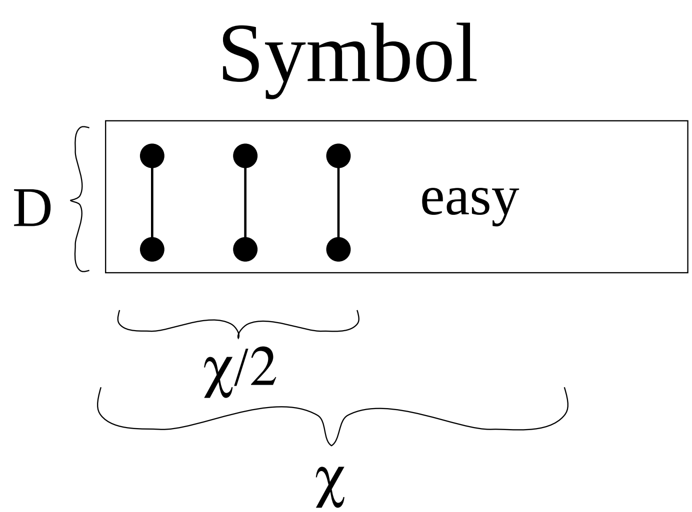
Step 1

Time $= \chi/8$, Space $= \infty$, any start config $\langle Q, \Pi \rangle$,

$$
\begin{aligned}
& \Pr_G [Prog \geq 3S] \\
= \quad & \sum_{\gamma \notin C} \Pr_G [Prog \geq 3S \mid G \text{ follows } \gamma] \times \Pr_G [G \text{ follows } \gamma] \\
+ \quad & \sum_{\gamma \in C} \Pr_G [Prog \geq 3S \mid G \text{ follows } \gamma] \times \Pr_G [G \text{ follows } \gamma] \\
\\
\leq \quad & \max_{\gamma \notin C} \Pr_G [Prog \geq 3S \mid G \text{ follows } \gamma] \qquad \times \qquad 1 \\
+ \quad & 1 \qquad \times \qquad \Pr_G [G \text{ follows } \gamma \in C] \\
\\
< \quad & \frac{1}{2} 2^{-S}
\end{aligned}
$$

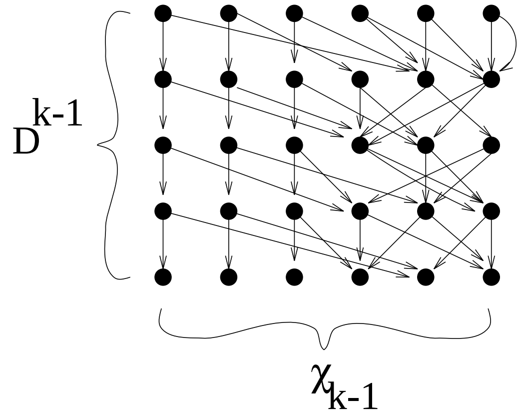
Step 3: Define Recursive Graph



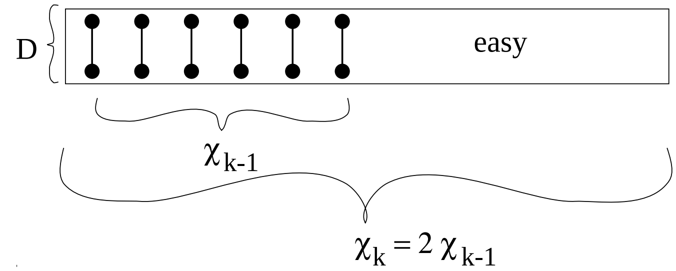


Step 3: Define Recursive Graph

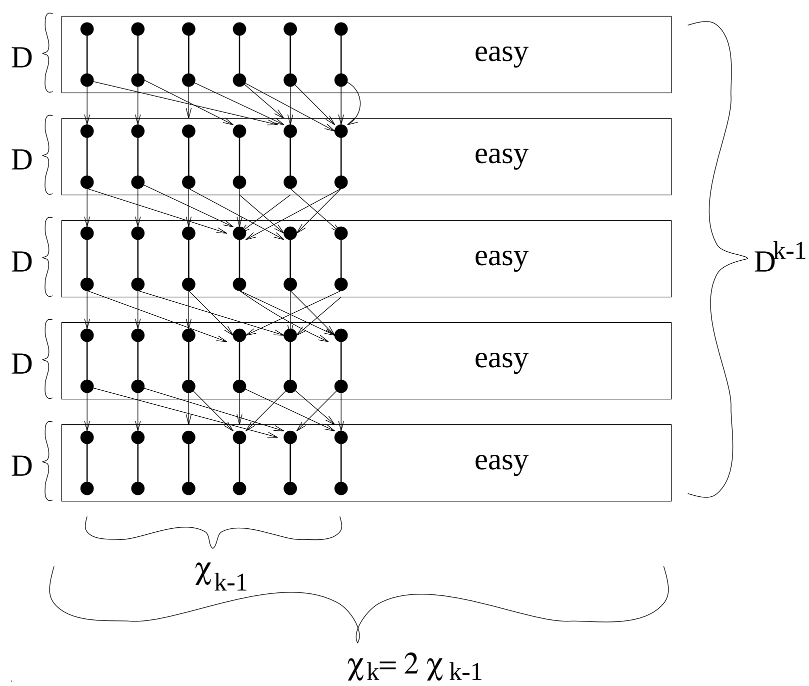
Recursive Layered Graph G_{k-1}



Boosting Graph



Recursive Layered Graph G_k



Step 4: Recursive Lower Bound

Induction Hypothesis:

- st-conn requires T_{k-1} time for G_{k-1}.

Consider G_k

In $\chi/8$ time can traverse	$\leq 3S$	"hard paths"
		$\approx$ "super nodes".
	$\leq 6S$	"super edges".

In T_k time can traverse $\leq \frac{6S}{\chi/8} \times T_k$.

Set to T_{k-1}.

Calculating the Bound

$$\begin{aligned} n &= \text{ width } \times \text{ height} \\ &= \left(\chi 2^K\right) \times \left(D^K\right) \end{aligned}$$

$$K = \frac{\log(n/\chi)}{2\log D}$$

$$\begin{aligned} T_K &= \left(\frac{\chi_k}{48S}\right) T_{k-1} \approx \left(\frac{\chi}{48S}\right)^K \\ &= 2^{\frac{\log(\chi/(48S))\log(n/\chi)}{\log D}} \end{aligned}$$

$$\begin{aligned} \chi/(48S) &= n/\chi \\ \chi &= \sqrt{48nS} \end{aligned}$$

$$\begin{aligned} T_K &= 2^{\frac{\log^2(\sqrt{n/(48S)})}{\log\log n}} \\ &= 2^{\Omega\left(\frac{\log^2(n/S)}{\log\log n}\right)} \end{aligned}$$

Strengthening the Model

- Let the JAG jump to the next node in a fixed ordering of the nodes.
  = General Model of Computation
- Let the JAG specify an indegree edge and back up it.
  within a poly factor in time is a General Model of Computation

Open Problems

$S \times T \in \Omega\left(n^{2-\epsilon}\right)$ for

- undirected graphs
- directed graphs