Simple Optimal Hitting Sets for Small-Success **RL**

William M. Hoza[1] David Zuckerman[2]

The University of Texas at Austin

October 7
FOCS 2018

[1] Supported by the NSF GRFP under Grant DGE-1610403 and by a Harrington Fellowship from UT Austin
[2] Supported by NSF Grant CCF-1526952, NSF Grant CCF-1705028, and a Simons Investigator Award (#409864)

Randomized log-space complexity classes

- Let L be a language

Randomized log-space complexity classes

- Let L be a language
- $L \in \mathbf{BPL}$ if there is a randomized log-space algorithm A that always halts such that

$$x \in L \implies \Pr[A(x) \text{ accepts}] \geq 2/3$$
$$x \notin L \implies \Pr[A(x) \text{ accepts}] \leq 1/3.$$

Randomized log-space complexity classes

- Let L be a language
- $L \in \mathbf{BPL}$ if there is a randomized log-space algorithm A that always halts such that

$$x \in L \implies \Pr[A(x) \text{ accepts}] \geq 2/3$$
$$x \notin L \implies \Pr[A(x) \text{ accepts}] \leq 1/3.$$

- $L \in \mathbf{RL}$ if there is a randomized log-space algorithm A that always halts such that

$$x \in L \implies \Pr[A(x) \text{ accepts}] \geq 1/2$$
$$x \notin L \implies \Pr[A(x) \text{ accepts}] = 0.$$

The power of randomness for small-space algorithms

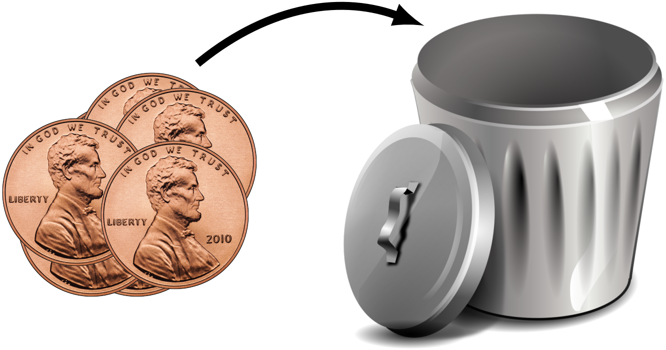
- $\mathbf{L} \subseteq \mathbf{RL} \subseteq \mathbf{BPL}$

The power of randomness for small-space algorithms

- $\mathbf{L} \subseteq \mathbf{RL} \subseteq \mathbf{BPL}$
- **Conjecture**: $\mathbf{L} = \mathbf{RL} = \mathbf{BPL}$

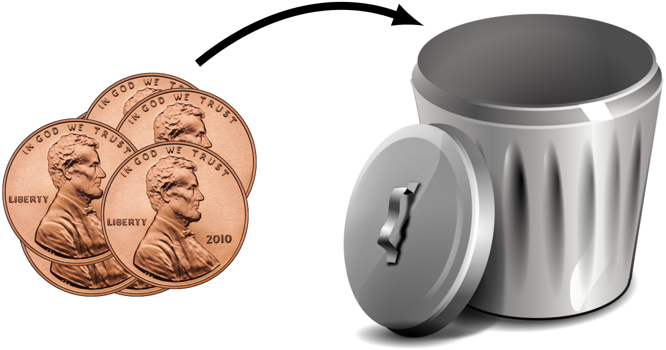
The power of randomness for small-space algorithms

- $\mathbf{L} \subseteq \mathbf{RL} \subseteq \mathbf{BPL}$
- **Conjecture**: $\mathbf{L} = \mathbf{RL} = \mathbf{BPL}$

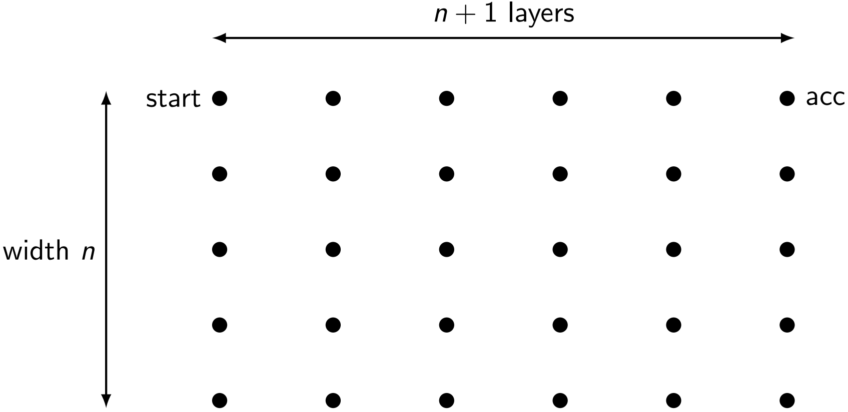


The power of randomness for small-space algorithms

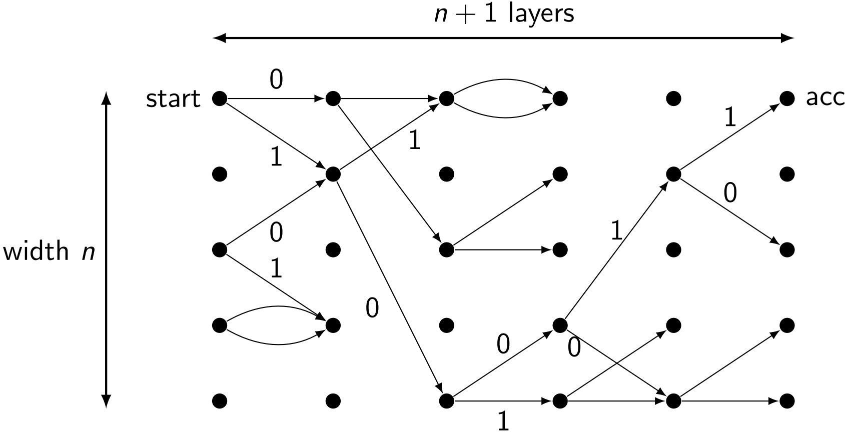
- $\mathbf{L} \subseteq \mathbf{RL} \subseteq \mathbf{BPL}$
- **Conjecture**: $\mathbf{L} = \mathbf{RL} = \mathbf{BPL}$



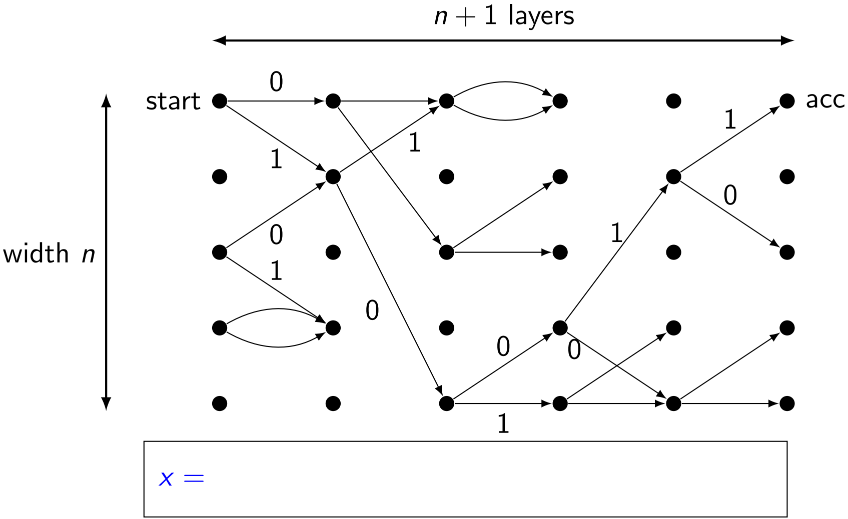
Read-once branching programs



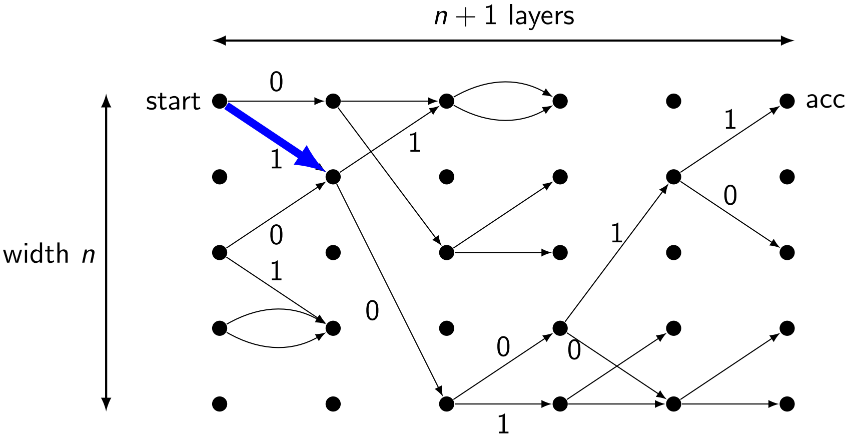
Read-once branching programs



Read-once branching programs

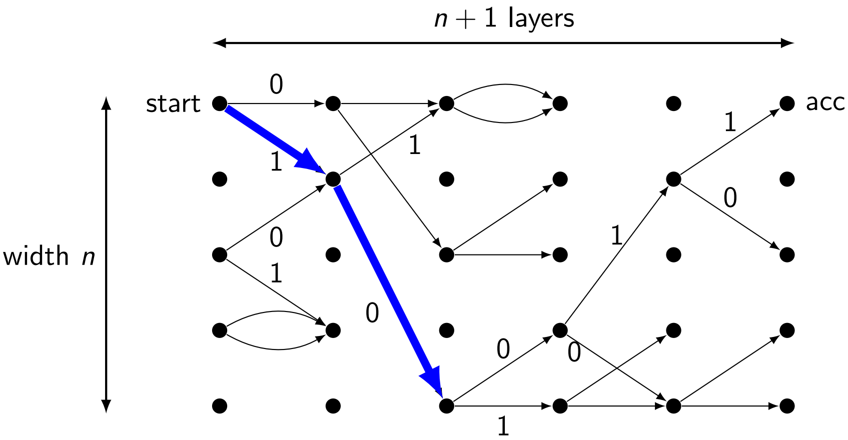


Read-once branching programs



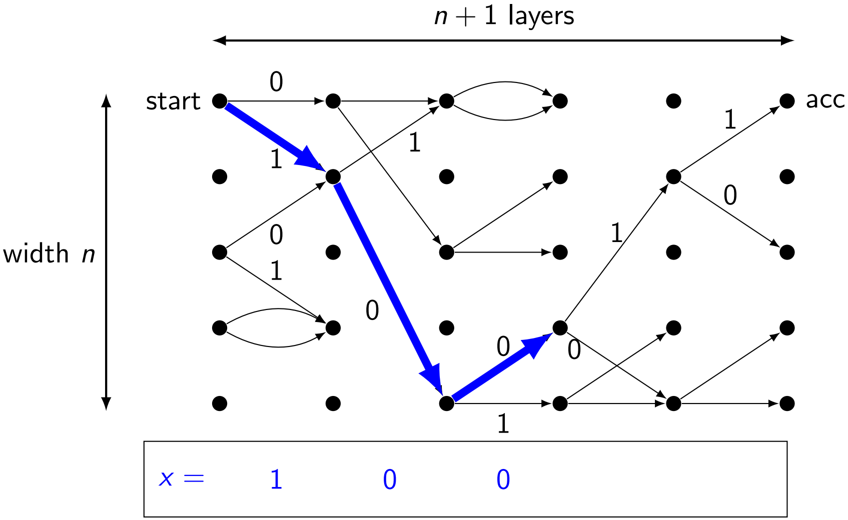
$x = \quad 1$

Read-once branching programs

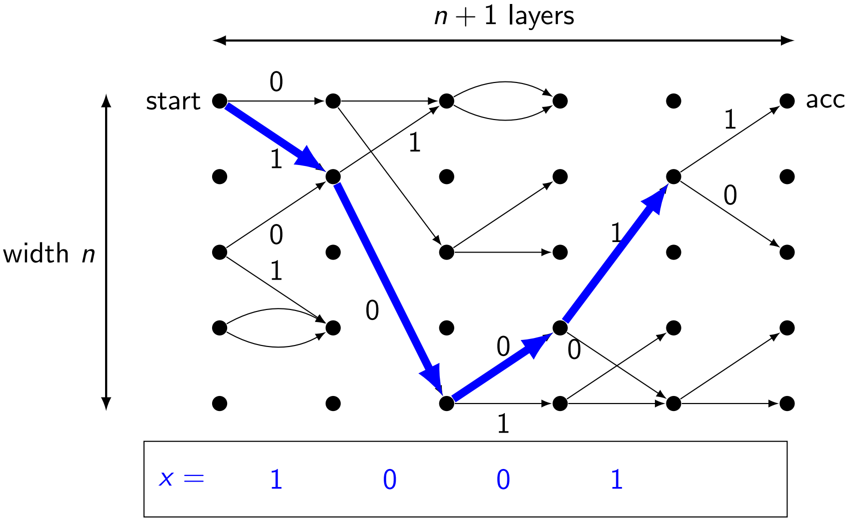


$x =$ 1 0

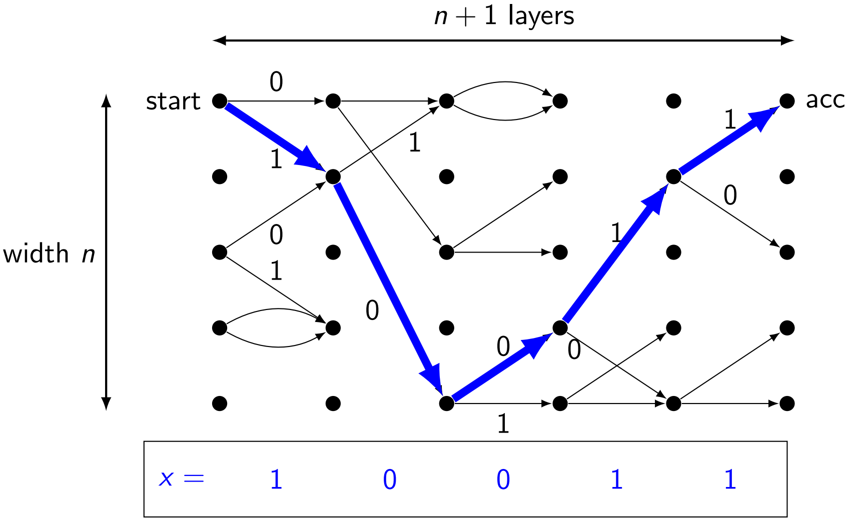
Read-once branching programs



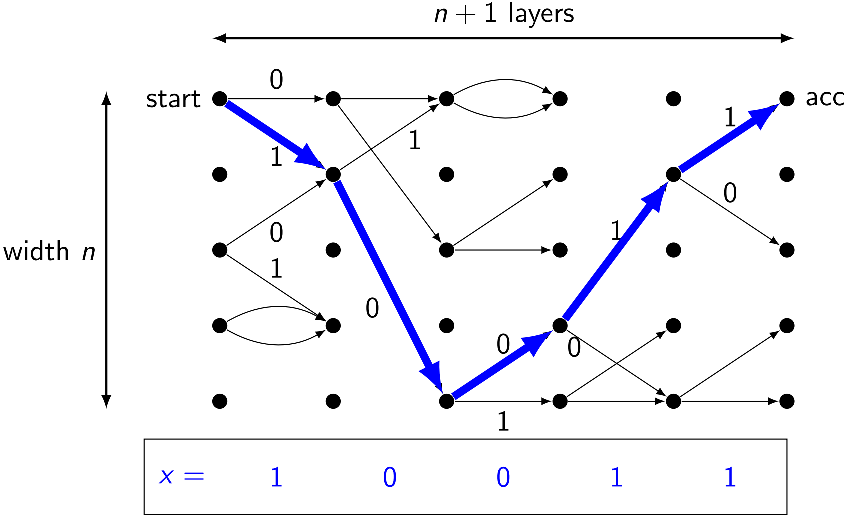
Read-once branching programs



Read-once branching programs

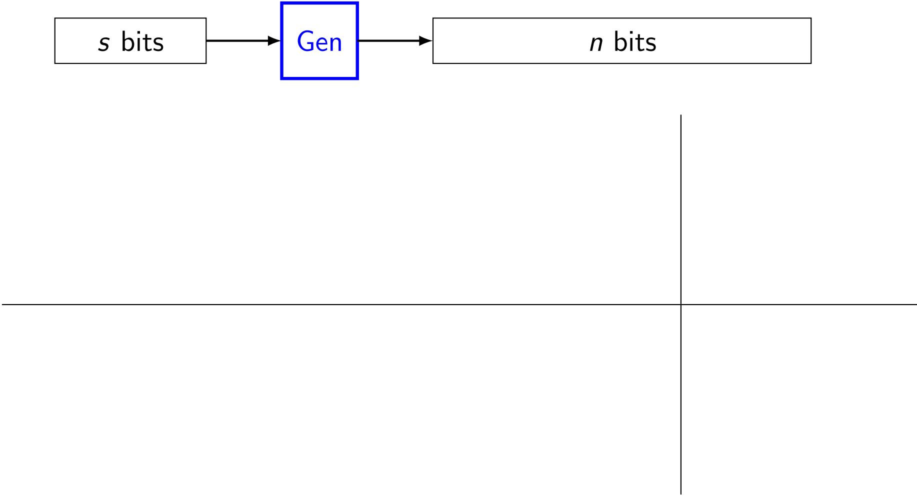


Read-once branching programs

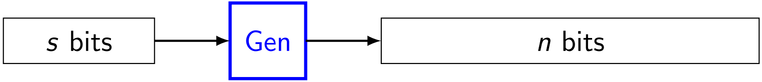


- Computes function $f : \{0,1\}^n \to \{0,1\}$

Fooling / Hitting ROBPs



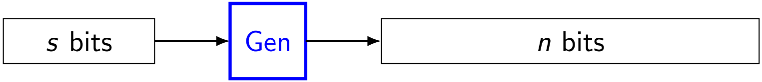
Fooling / Hitting ROBPs



Pseudorandom generator: For every width-n ROBP,

$$|\Pr_x[f(x) = 1] - \Pr_z[f(\mathrm{Gen}(z)) = 1]| \leq \varepsilon$$

Fooling / Hitting ROBPs

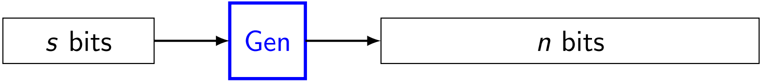


Pseudorandom generator: For every width-n ROBP,

$$|\Pr_x[f(x)=1] - \Pr_z[f(\mathrm{Gen}(z))=1]| \le \varepsilon$$

Suitable for derandomizing **BPL**

Fooling / Hitting ROBPs



Pseudorandom generator: For every width-n ROBP,

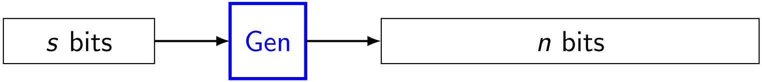
$$|\Pr_x[f(x) = 1] - \Pr_z[f(\mathrm{Gen}(z)) = 1]| \leq \varepsilon$$

Suitable for derandomizing **BPL**

Hitting set generator: For every width-n ROBP,

$$\Pr_x[f(x) = 1] \geq \varepsilon \implies \exists z, f(\mathrm{Gen}(z)) = 1$$

Fooling / Hitting ROBPs



Pseudorandom generator: For every width-n ROBP, $$\left|\Pr_x[f(x)=1]-\Pr_z[f(\mathrm{Gen}(z))=1]\right|\le\varepsilon$$	Suitable for derandomizing **BPL**
Hitting set generator: For every width-n ROBP, $$\Pr_x[f(x)=1]\ge\varepsilon \implies \exists z, f(\mathrm{Gen}(z))=1$$	Suitable for derandomizing **RL**

Prior generators and main result

- Nonconstructive: PRG with seed length $O(\log n + \log(1/\varepsilon))$

Prior generators and main result

- Nonconstructive: PRG with seed length $O(\log n + \log(1/\varepsilon))$
- Babai, Nisan, Szegedy 1989: PRG with seed length

$$2^{O(\sqrt{\log n})} \cdot \log(1/\varepsilon)$$

Prior generators and main result

- Nonconstructive: PRG with seed length $O(\log n + \log(1/\varepsilon))$
- Babai, Nisan, Szegedy 1989: PRG with seed length

$$2^{O(\sqrt{\log n})} \cdot \log(1/\varepsilon)$$

- Nisan 1990: PRG with seed length

$$O(\log^2 n + \log(1/\varepsilon)\log n)$$

Prior generators and main result

- Nonconstructive: PRG with seed length $O(\log n + \log(1/\varepsilon))$
- Babai, Nisan, Szegedy 1989: PRG with seed length

$$2^{O(\sqrt{\log n})} \cdot \log(1/\varepsilon)$$

- Nisan 1990: PRG with seed length

$$O(\log^2 n + \log(1/\varepsilon) \log n)$$

- Braverman, Cohen, Garg 2018: HSG with seed length

$$\widetilde{O}(\log^2 n + \log(1/\varepsilon))$$

Prior generators and main result

- Nonconstructive: PRG with seed length $O(\log n + \log(1/\varepsilon))$
- Babai, Nisan, Szegedy 1989: PRG with seed length

$$2^{O(\sqrt{\log n})} \cdot \log(1/\varepsilon)$$

- Nisan 1990: PRG with seed length

$$O(\log^2 n + \log(1/\varepsilon) \log n)$$

- Braverman, Cohen, Garg 2018: HSG with seed length

$$\widetilde{O}(\log^2 n + \log(1/\varepsilon))$$

- **This work**: HSG with seed length

$$O(\log^2 n + \log(1/\varepsilon))$$

Comparison with [BCG '18]

- Our construction and analysis are simple

Comparison with [BCG '18]

- Our construction and analysis are simple

This work

Hitting Set Generator

Suitable for **RL**

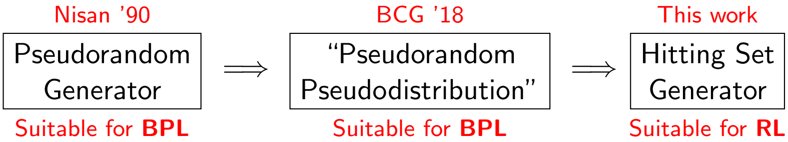
Comparison with [BCG '18]

- Our construction and analysis are simple

Nisan '90		This work
Pseudorandom Generator	$\Longrightarrow$	Hitting Set Generator
Suitable for **BPL**		Suitable for **RL**

Comparison with [BCG '18]

- Our construction and analysis are simple



Structural lemma for ROBPs

- Let f be a width-n, length-n ROBP

Structural lemma for ROBPs

- Let f be a width-n, length-n ROBP
- Assume $\Pr[\text{accept}] = \varepsilon \ll 1/n^3$

Structural lemma for ROBPs

- Let f be a width-n, length-n ROBP
- Assume $\Pr[\text{accept}] = \varepsilon \ll 1/n^3$
- **Lemma**: There is a vertex u so that

$$\Pr[\text{reach } u] \geq \frac{1}{2n^3} \quad \text{and} \quad \Pr[\text{accept} \mid \text{reach } u] \geq \varepsilon \boldsymbol{n}.$$

Proof of lemma ($\exists u, \Pr[u] \geq \frac{1}{2n^3} \wedge \Pr[\text{acc} \mid u] \geq \varepsilon n$)

- Say u is a milestone if $\Pr[\text{accept} \mid \text{reach } u] \in [\varepsilon n, 2\varepsilon n]$

Proof of lemma $(\exists u, \Pr[u] \geq \frac{1}{2n^3} \wedge \Pr[\text{acc} \mid u] \geq \varepsilon n)$

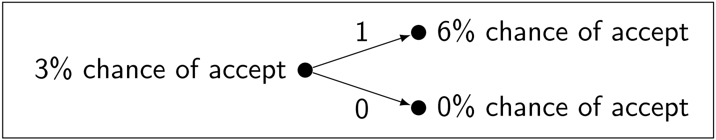
- Say u is a milestone if $\Pr[\text{accept} \mid \text{reach } u] \in [\varepsilon n, 2\varepsilon n]$
- Claim: Every accepting path passes through a milestone

Proof of lemma $(\exists u, \Pr[u] \geq \frac{1}{2n^3} \wedge \Pr[\text{acc} \mid u] \geq \varepsilon n)$

- Say u is a milestone if $\Pr[\text{accept} \mid \text{reach } u] \in [\varepsilon n, 2\varepsilon n]$
- Claim: Every accepting path passes through a milestone
  - Proof: Probability of acceptance at most doubles in each step

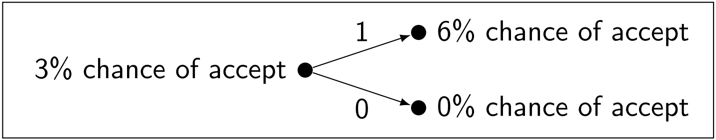
Proof of lemma $(\exists u, \Pr[u] \geq \frac{1}{2n^3} \wedge \Pr[\text{acc} \mid u] \geq \varepsilon n)$

- Say u is a milestone if $\Pr[\text{accept} \mid \text{reach } u] \in [\varepsilon n, 2\varepsilon n]$
- Claim: Every accepting path passes through a milestone
  - Proof: Probability of acceptance at most doubles in each step



Proof of lemma $(\exists u, \Pr[u] \geq \frac{1}{2n^3} \wedge \Pr[\text{acc} \mid u] \geq \varepsilon n)$

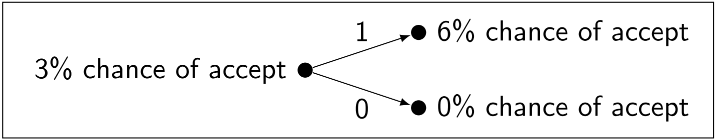
- Say u is a milestone if $\Pr[\text{accept} \mid \text{reach } u] \in [\varepsilon n, 2\varepsilon n]$
- Claim: Every accepting path passes through a milestone
  - Proof: Probability of acceptance at most doubles in each step



- $\varepsilon = \Pr[\text{accept}] \leq \sum_{u \text{ milestone}} \Pr[\text{reach } u \text{ and accept}]$

Proof of lemma $(\exists u, \Pr[u] \geq \frac{1}{2n^3} \wedge \Pr[\mathrm{acc} \mid u] \geq \varepsilon n)$

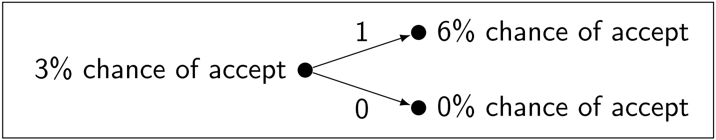
- Say u is a milestone if $\Pr[\text{accept} \mid \text{reach } u] \in [\varepsilon n, 2\varepsilon n]$
- Claim: Every accepting path passes through a milestone
  - Proof: Probability of acceptance at most doubles in each step



- $$\varepsilon = \Pr[\text{accept}] \leq \sum_{u \text{ milestone}} \Pr[\text{reach } u \text{ and accept}] \leq \sum_{u \text{ milestone}} \Pr[\text{reach } u] \cdot 2\varepsilon n$$

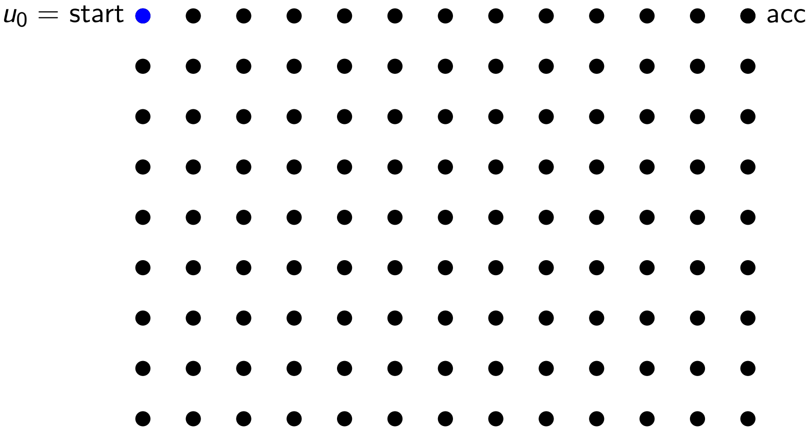
Proof of lemma $(\exists u, \Pr[u] \geq \frac{1}{2n^3} \wedge \Pr[\text{acc} \mid u] \geq \varepsilon n)$

- Say u is a milestone if $\Pr[\text{accept} \mid \text{reach } u] \in [\varepsilon n, 2\varepsilon n]$
- Claim: Every accepting path passes through a milestone
  - Proof: Probability of acceptance at most doubles in each step



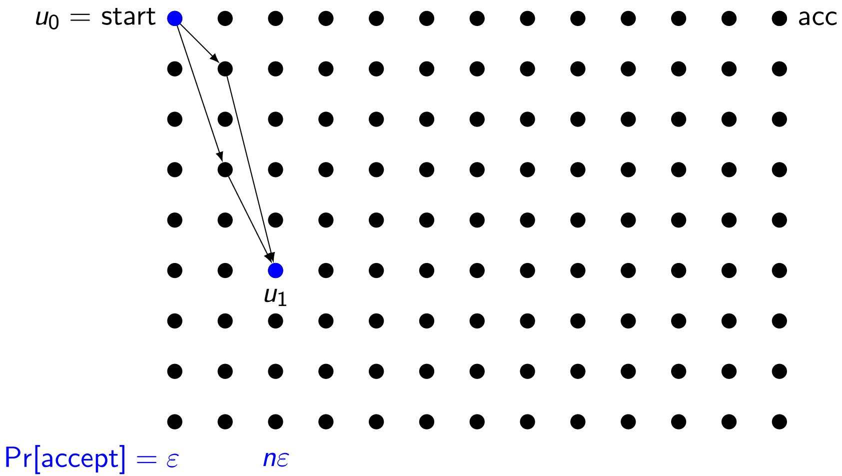
- $$\varepsilon = \Pr[\text{accept}] \leq \sum_{u \text{ milestone}} \Pr[\text{reach } u \text{ and accept}]$$
  $$\leq \sum_{u \text{ milestone}} \Pr[\text{reach } u] \cdot 2\varepsilon n$$
- $\#$ milestones $\leq n^2$, so for some milestone u, $\Pr[\text{reach } u] \geq \frac{1}{2n^3}$ $\square$

Iterating the structural lemma

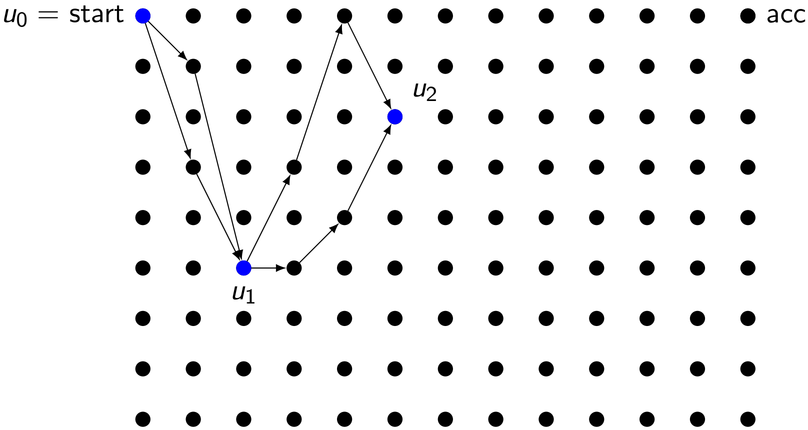


$\Pr[\text{accept}] = \varepsilon$

Iterating the structural lemma

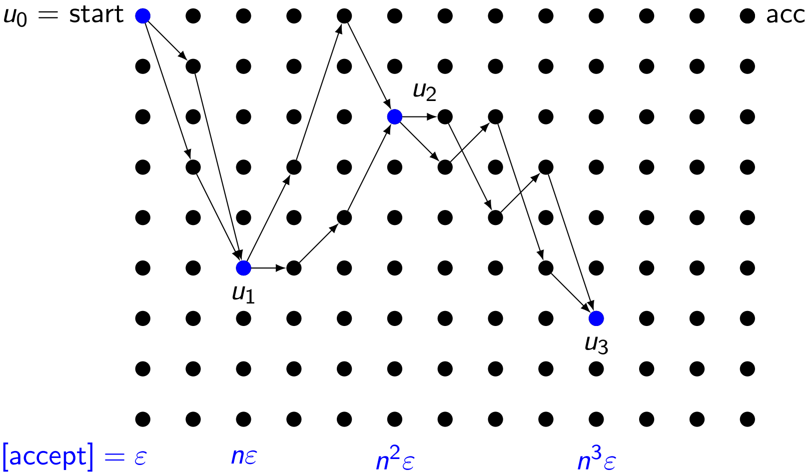


Iterating the structural lemma

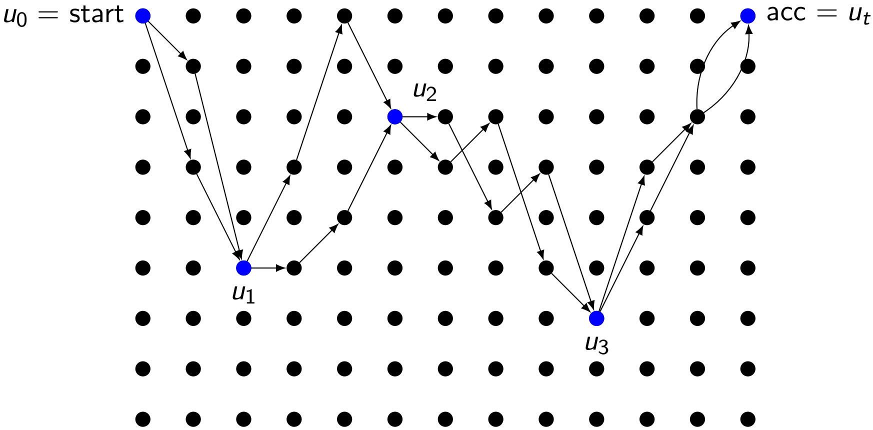


$\Pr[\text{accept}] = \varepsilon$ $\quad n\varepsilon \quad n^2\varepsilon$

Iterating the structural lemma



Iterating the structural lemma



Pr[accept] $= \varepsilon$ $\quad n\varepsilon \quad n^2\varepsilon \quad n^3\varepsilon \quad n^t\varepsilon = 1$

Idea of our HSG

- Use Nisan's generator for each individual hop $u_i \to u_{i+1}$

Idea of our HSG

- Use Nisan's generator for each individual hop $u_i \to u_{i+1}$
- Use a "hitter" to recycle the seed of Nisan's generator from one hop to the next

Hitters (equivalent to dispersers)

- Assume query access to unknown $E \subseteq \{0,1\}^m$ with density$(E) \geq \theta$

Hitters (equivalent to dispersers)

- Assume query access to unknown $E \subseteq \{0,1\}^m$ with density$(E) \geq \theta$
- **Theorem** (BGG '93): Algorithm that outputs some $z \in E$ with probability $1 - \delta$

Hitters (equivalent to dispersers)

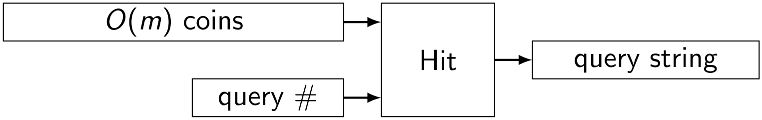
- Assume query access to unknown $E \subseteq \{0,1\}^m$ with density$(E) \geq \theta$
- **Theorem** (BGG '93): Algorithm that outputs some $z \in E$ with probability $1 - \delta$
  - # queries: $O(\theta^{-1} \cdot \log(1/\delta))$

Hitters (equivalent to dispersers)

- Assume query access to unknown $E \subseteq \{0,1\}^m$ with density$(E) \geq \theta$
- **Theorem** (BGG '93): Algorithm that outputs some $z \in E$ with probability $1 - \delta$
  - # queries: $O(\theta^{-1} \cdot \log(1/\delta))$
  - # random bits: $O(m + \log(1/\delta))$

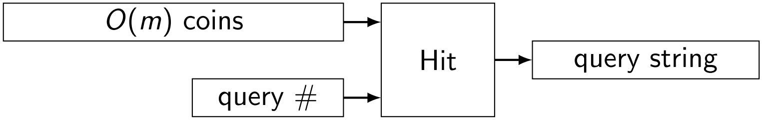
Hitters (equivalent to dispersers)

- Assume query access to unknown $E \subseteq \{0,1\}^m$ with density$(E) \geq \theta$
- **Theorem** (BGG '93): Algorithm that outputs some $z \in E$ with probability $1 - \delta$
  - # queries: $O(\theta^{-1} \cdot \log(1/\delta))$
  - # random bits: $O(m + \log(1/\delta))$



Hitters (equivalent to dispersers)

- Assume query access to unknown $E \subseteq \{0,1\}^m$ with density$(E) \geq \theta$
- **Theorem** (BGG '93): Algorithm that outputs some $z \in E$ with probability $1 - \delta$
  - # queries: $O(\theta^{-1} \cdot \log(1/\delta))$
  - # random bits: $O(m + \log(1/\delta))$



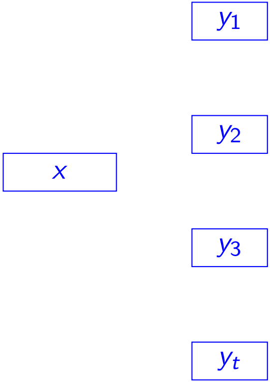
- For any E with density$(E) \geq \theta$,

$$\Pr_x[\exists y, \mathrm{Hit}(x,y) \in E] \geq 1 - \delta$$

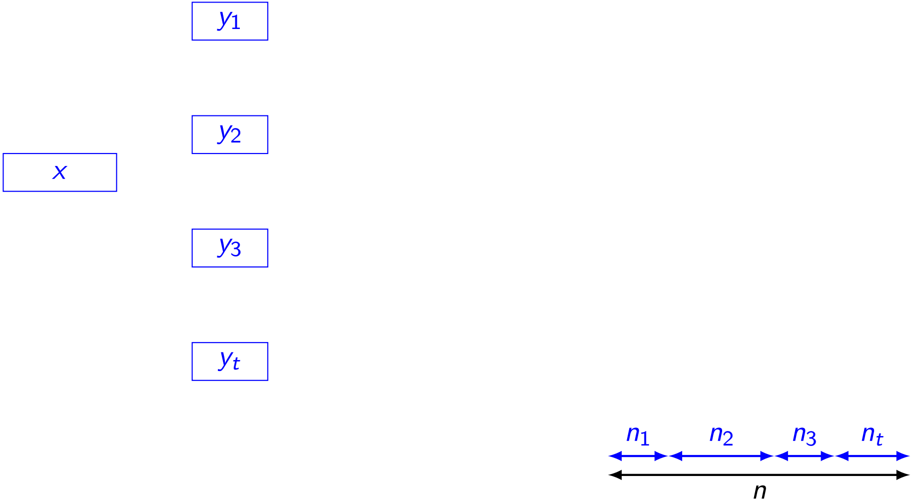
Our HSG

x

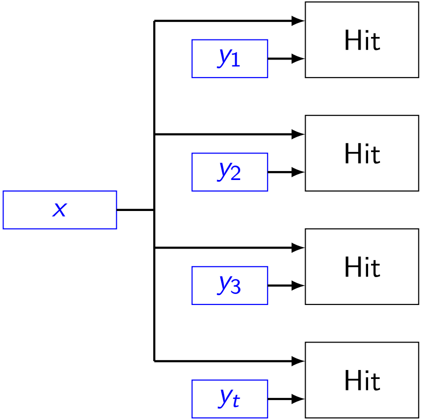
Our HSG

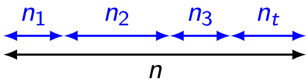


Our HSG

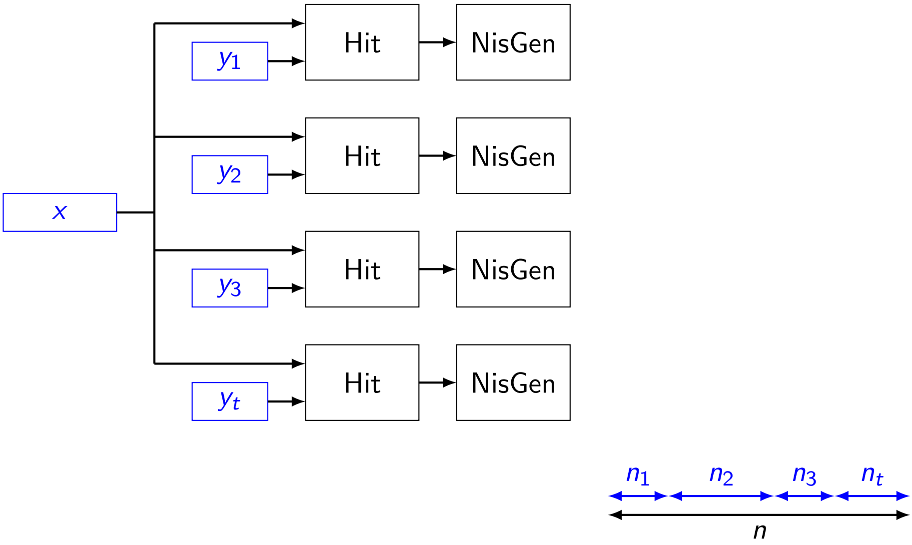


Our HSG



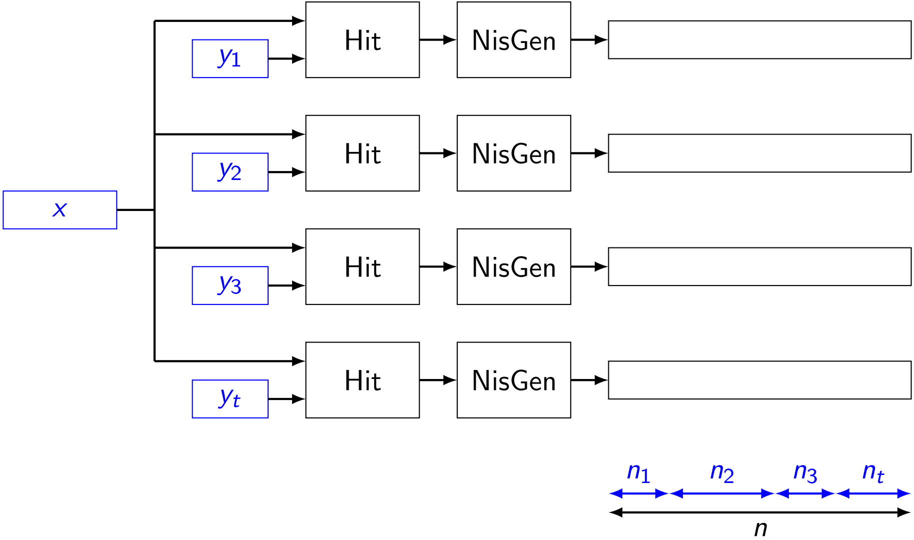


Our HSG



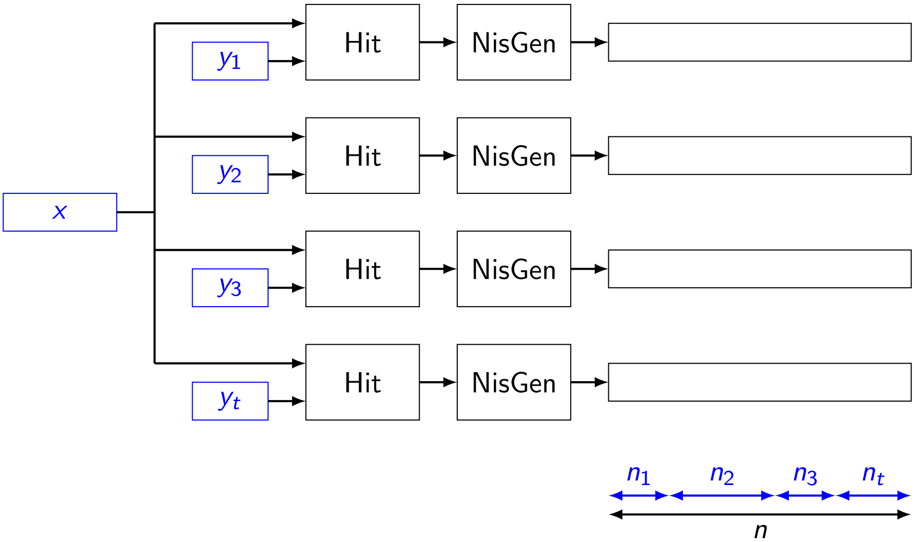
Hit
NisGen
y_1
Hit
NisGen
y_2
x
Hit
NisGen
y_3
Hit
NisGen
y_t
n_1
n_2
n_3
n_t
n

Our HSG



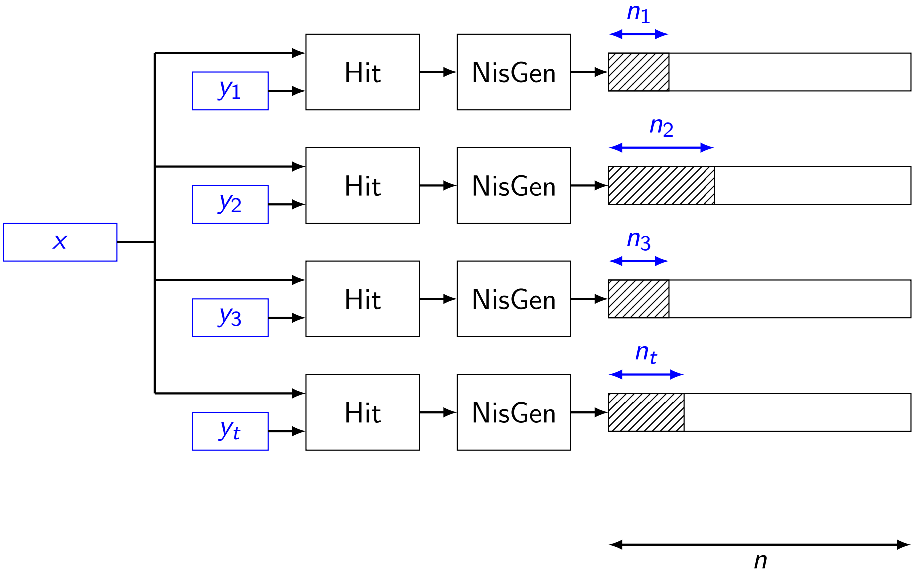
y_1
Hit
NisGen
y_2
Hit
NisGen
x
y_3
Hit
NisGen
y_t
Hit
NisGen
n_1
n_2
n_3
n_t
n

Our HSG



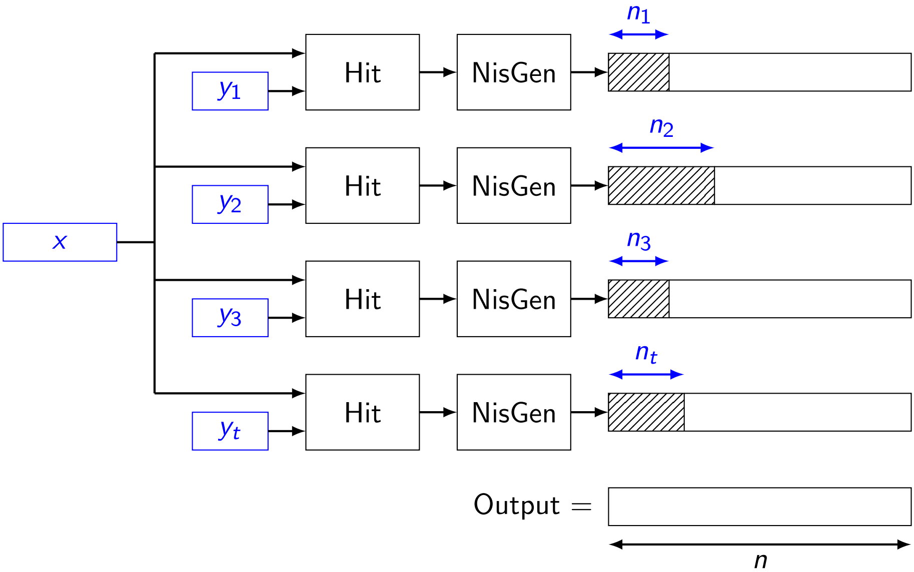
Hit
NisGen
y_1
Hit
NisGen
y_2
x
Hit
NisGen
y_3
Hit
NisGen
y_t
n_1
n_2
n_3
n_t
n

Our HSG



n_1
y_1
Hit
NisGen
n_2
y_2
Hit
NisGen
x
n_3
y_3
Hit
NisGen
n_t
y_t
Hit
NisGen
n

Our HSG


n_1
n_2
n_3
n_t
x
y_1
y_2
y_3
y_t
Hit
Hit
Hit
Hit
NisGen
NisGen
NisGen
NisGen
Output =
n

Our HSG in symbols

- For numbers $n_1, \ldots, n_t$ with $n_1 + \cdots + n_t = n$:

$$\begin{aligned} &\mathrm{Gen}(x, y_1, \ldots, y_t, n_1, \ldots, n_t) = \\ &\quad \mathrm{NisGen}(\mathrm{Hit}(x, y_1))|_{n_1} \circ \cdots \circ \mathrm{NisGen}(\mathrm{Hit}(x, y_t))|_{n_t} \in \{0,1\}^n \end{aligned}$$

Our HSG in symbols

- For numbers $n_1, \ldots, n_t$ with $n_1 + \cdots + n_t = n$:

$$\mathrm{Gen}(x, y_1, \ldots, y_t, n_1, \ldots, n_t) = \mathrm{NisGen}(\mathrm{Hit}(x, y_1))|_{n_1} \circ \cdots \circ \mathrm{NisGen}(\mathrm{Hit}(x, y_t))|_{n_t} \in \{0,1\}^n$$

- Here $\circ$ = concatenation, $|_r$ = first r bits

Our HSG in symbols

- For numbers $n_1, \ldots, n_t$ with $n_1 + \cdots + n_t = n$:

$$\mathrm{Gen}(x, y_1, \ldots, y_t, n_1, \ldots, n_t) = \mathrm{NisGen}(\mathrm{Hit}(x, y_1))|_{n_1} \circ \cdots \circ \mathrm{NisGen}(\mathrm{Hit}(x, y_t))|_{n_t} \in \{0,1\}^n$$

- Here $\circ$ = concatenation, $|_r$ = first r bits
- $|x| = O(\log^2 n)$, $|y_i| = O(\log n)$, $t = \frac{\log(1/\varepsilon)}{\log n}$

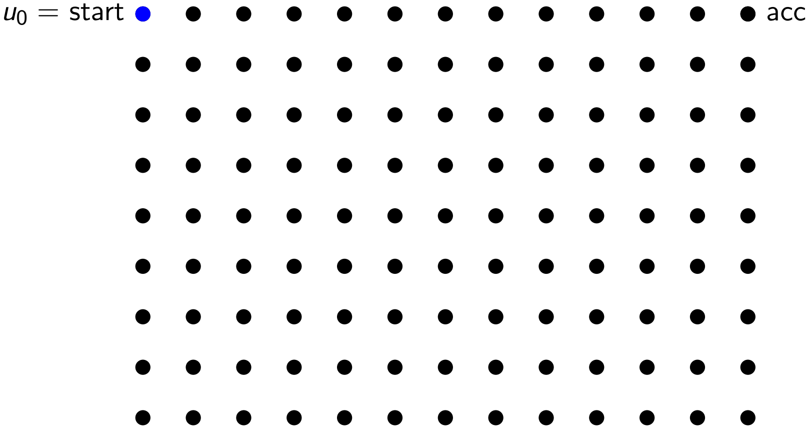
Our HSG in symbols

- For numbers $n_1, \ldots, n_t$ with $n_1 + \cdots + n_t = n$:

$$\mathrm{Gen}(x, y_1, \ldots, y_t, n_1, \ldots, n_t) = \mathrm{NisGen}(\mathrm{Hit}(x, y_1))|_{n_1} \circ \cdots \circ \mathrm{NisGen}(\mathrm{Hit}(x, y_t))|_{n_t} \in \{0,1\}^n$$

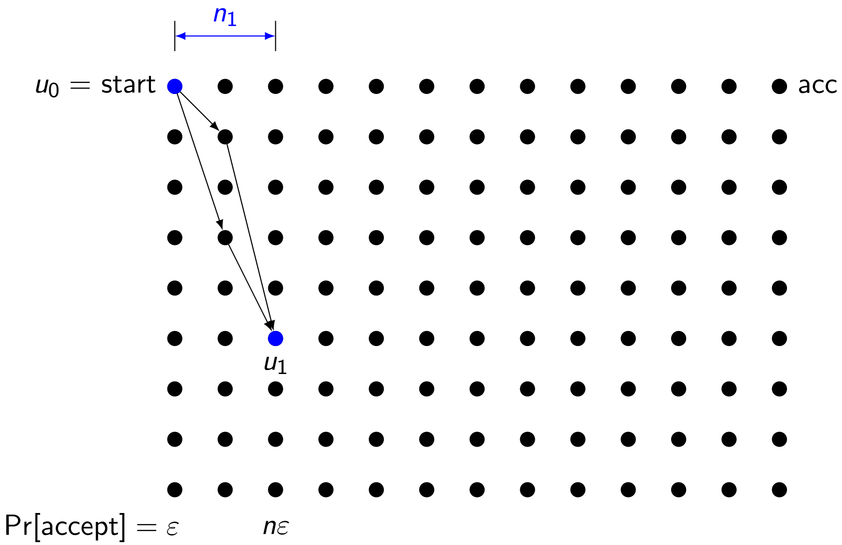
- Here $\circ$ = concatenation, $|_r$ = first r bits
- $|x| = O(\log^2 n)$, $|y_i| = O(\log n)$, $t = \frac{\log(1/\varepsilon)}{\log n}$
- So seed length $= O(\log^2 n + \log(1/\varepsilon))$

Proof of correctness of our HSG

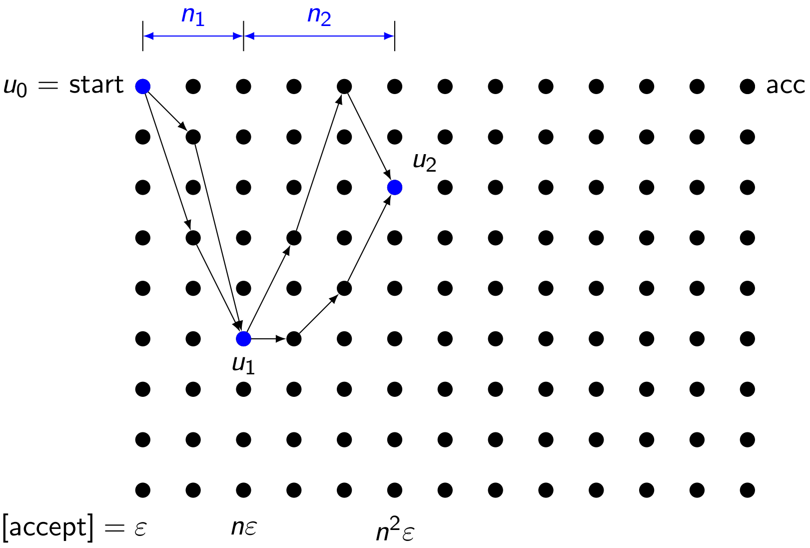


$\Pr[\text{accept}] = \varepsilon$

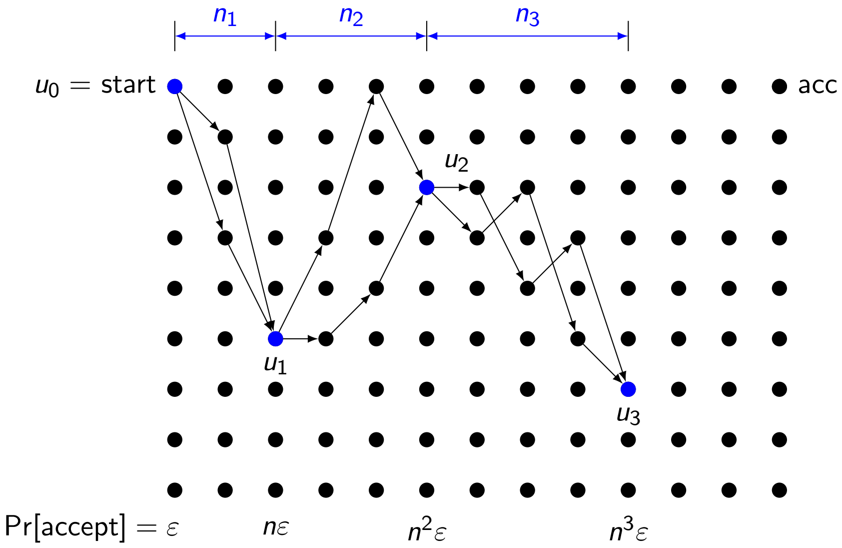
Proof of correctness of our HSG



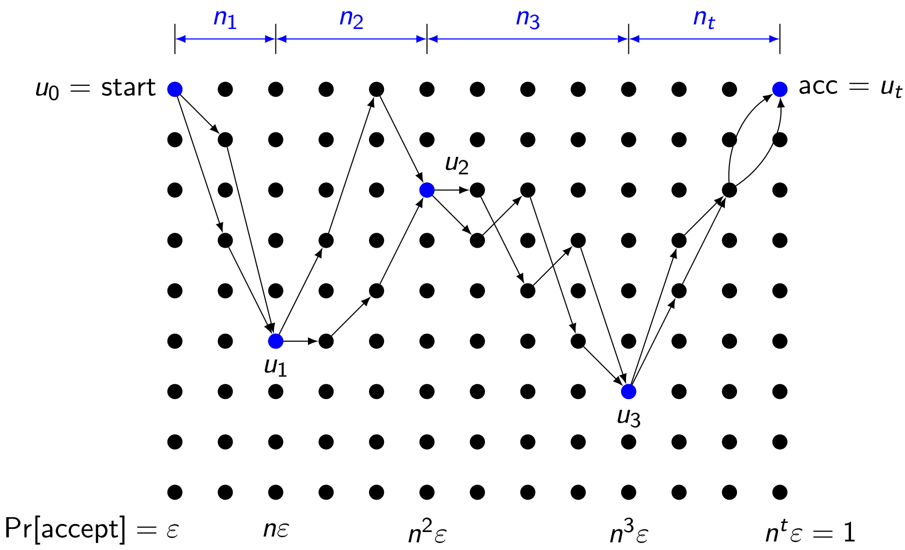
Proof of correctness of our HSG



Proof of correctness of our HSG



Proof of correctness of our HSG



Proof of correctness of our HSG (continued)

- Define $E_i \subseteq \{0,1\}^m$ by

$$E_i = \{z \mid \text{start at } u_{i-1}, \text{ read NisGen}(z) \implies \text{reach } u_i\}$$

Proof of correctness of our HSG (continued)

- Define $E_i \subseteq \{0,1\}^m$ by

$$E_i = \{z \mid \text{start at } u_{i-1}, \text{ read NisGen}(z) \implies \text{ reach } u_i\}$$

- $\Pr[\text{reach } u_i \mid \text{reach } u_{i-1}] \geq \frac{1}{2n^3} \implies \text{density}(E_i) > \frac{1}{4n^3}$

Proof of correctness of our HSG (continued)

- Define $E_i \subseteq \{0,1\}^m$ by

$$E_i = \{z \mid \text{start at } u_{i-1}, \text{ read NisGen}(z) \implies \text{ reach } u_i\}$$

- $\Pr[\text{reach } u_i \mid \text{reach } u_{i-1}] \geq \frac{1}{2n^3} \implies \text{density}(E_i) > \frac{1}{4n^3}$
- Hitter property: $\Pr_x[\exists y, \text{Hit}(x,y) \in E_i] > 1 - \frac{1}{t}$

Proof of correctness of our HSG (continued)

- Define $E_i \subseteq \{0,1\}^m$ by

$$E_i = \{z \mid \text{start at } u_{i-1}, \text{ read NisGen}(z) \implies \text{ reach } u_i\}$$

- $\Pr[\text{reach } u_i \mid \text{reach } u_{i-1}] \geq \frac{1}{2n^3} \implies \text{density}(E_i) > \frac{1}{4n^3}$
- Hitter property: $\Pr_x[\exists y, \text{Hit}(x,y) \in E_i] > 1 - \frac{1}{t}$
- Union bound: There is one x so that for all i,

$$\exists y_i, \text{Hit}(x, y_i) \in E_i.$$

Proof of correctness of our HSG (continued)

- Define $E_i \subseteq \{0,1\}^m$ by

$$E_i = \{z \mid \text{start at } u_{i-1}, \text{ read NisGen}(z) \implies \text{ reach } u_i\}$$

- $\Pr[\text{reach } u_i \mid \text{reach } u_{i-1}] \geq \frac{1}{2n^3} \implies \text{density}(E_i) > \frac{1}{4n^3}$
- Hitter property: $\Pr_x[\exists y, \text{Hit}(x,y) \in E_i] > 1 - \frac{1}{t}$
- Union bound: There is one x so that for all i,

$$\exists y_i, \text{Hit}(x, y_i) \in E_i.$$

- $f(\text{Gen}(x, y_1, \ldots, y_t, n_1, \ldots, n_t)) = 1$ $\square$

Additional results

- **Theorem**:

$$(\varepsilon\text{-success } \mathbf{RL}) \subseteq \mathbf{DSPACE}(\log^{3/2} n + \log n \log\log(1/\varepsilon))$$

Additional results

- **Theorem**:

$$(\varepsilon\text{-success } \mathbf{RL}) \subseteq \mathbf{DSPACE}(\log^{3/2} n + \log n \log\log(1/\varepsilon))$$

- **Theorem**: For ROBPs with width n and length polylog n, HSG with seed length $O(\log(n/\varepsilon))$

Additional results

- **Theorem**:

$$(\varepsilon\text{-success } \mathbf{RL}) \subseteq \mathbf{DSPACE}(\log^{3/2} n + \log n \log\log(1/\varepsilon))$$

- **Theorem**: For ROBPs with width n and length polylog n, HSG with seed length $O(\log(n/\varepsilon))$

- **Theorem**: For any $r = r(n)$, for any constant c,

$$(\mathbf{RL} \text{ with } r \text{ coins}) \subseteq \left(\mathbf{NL} \text{ with } \frac{r}{\log^c n} \text{ nondeterministic bits}\right)$$

Open questions

- **Conjecture**: For any $r = r(n)$, for any constant c,

$$(\textbf{BPL} \text{ with } r \text{ coins}) = \left(\textbf{BPL} \text{ with } \frac{r}{\log^c n} \text{ coins}\right)$$

Open questions

- **Conjecture**: For any $r = r(n)$, for any constant c,

$$(\textbf{BPL with } r \text{ coins}) = \left(\textbf{BPL with } \frac{r}{\log^c n} \text{ coins}\right)$$

  - True for $r \leq 2^{\log^{0.99} n}$ by Nisan-Zuckerman

Open questions

- **Conjecture**: For any $r = r(n)$, for any constant c,

$$(\textbf{BPL with } r \text{ coins}) = \left(\textbf{BPL with } \frac{r}{\log^c n} \text{ coins}\right)$$

  - True for $r \leq 2^{\log^{0.99} n}$ by Nisan-Zuckerman

- ACR '96: Explicit HSG for circuits $\implies$ $\mathbf{P} = \mathbf{BPP}$. Similar theorem for **BPL**?

Open questions

- **Conjecture**: For any $r = r(n)$, for any constant c,

$$(\mathbf{BPL} \text{ with } r \text{ coins}) = \left(\mathbf{BPL} \text{ with } \frac{r}{\log^c n} \text{ coins}\right)$$

  - True for $r \leq 2^{\log^{0.99} n}$ by Nisan-Zuckerman

- ACR '96: Explicit HSG for circuits $\implies$ $\mathbf{P} = \mathbf{BPP}$. Similar theorem for **BPL**?

- Thanks! Questions?