Derandomizing using Nisan's PRG

- Need to fool $O(\log n)$ space algorithm and require poly(n) length bitstring

- Require $l, t = \Omega(\log n)$

- Seed of size $O(\log^2 n)$

- Need to evaluate $O(\log n)$ hash functions to retrieve a block

- We use $\Omega(n^{1-2/p})$ bits to store the state

- Can we use a larger seed for the PRG to decrease $O(\log n)$ hash evaluations?

35

Derandomizing using Nisan's PRG

- Need to fool $O(\log n)$ space algorithm and require poly(n) length bitstring
  - Require $l, t = \Omega(\log n)$
  - Seed of size $O(\log^2 n)$
  - Need to evaluate $O(\log n)$ hash functions to retrieve a block
- We use $\Omega(n^{1-2/p})$ bits to store the state
  - Can we use a larger seed for the PRG to decrease $O(\log n)$ hash evaluations?

HashPRG : Our Construction