Preserving Properties via Nisan's PRG

- Consider the following algorithm:

- Uses w bits of space to store its state ($\leq 2^w$ states)

- Makes a single-pass on the uniform random string

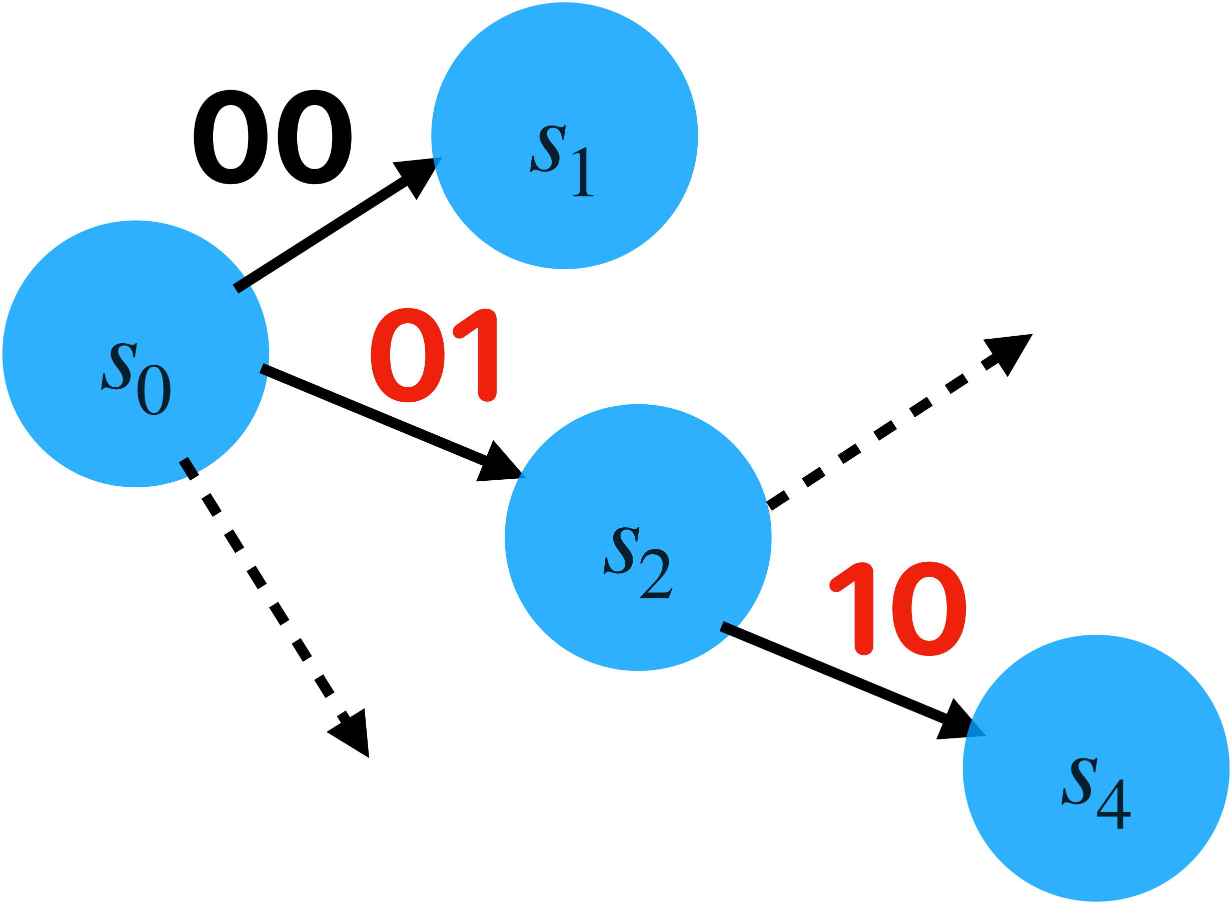
- Updates its state according to an arbitrary state transition table

- Nisan gave a PRG to "fool" such algorithms by replacing a fully random string with pseudorandom string generated using a **short uniform random seed**

- Indyk gave a recipe to fool turnstile streaming algorithms using such PRGs

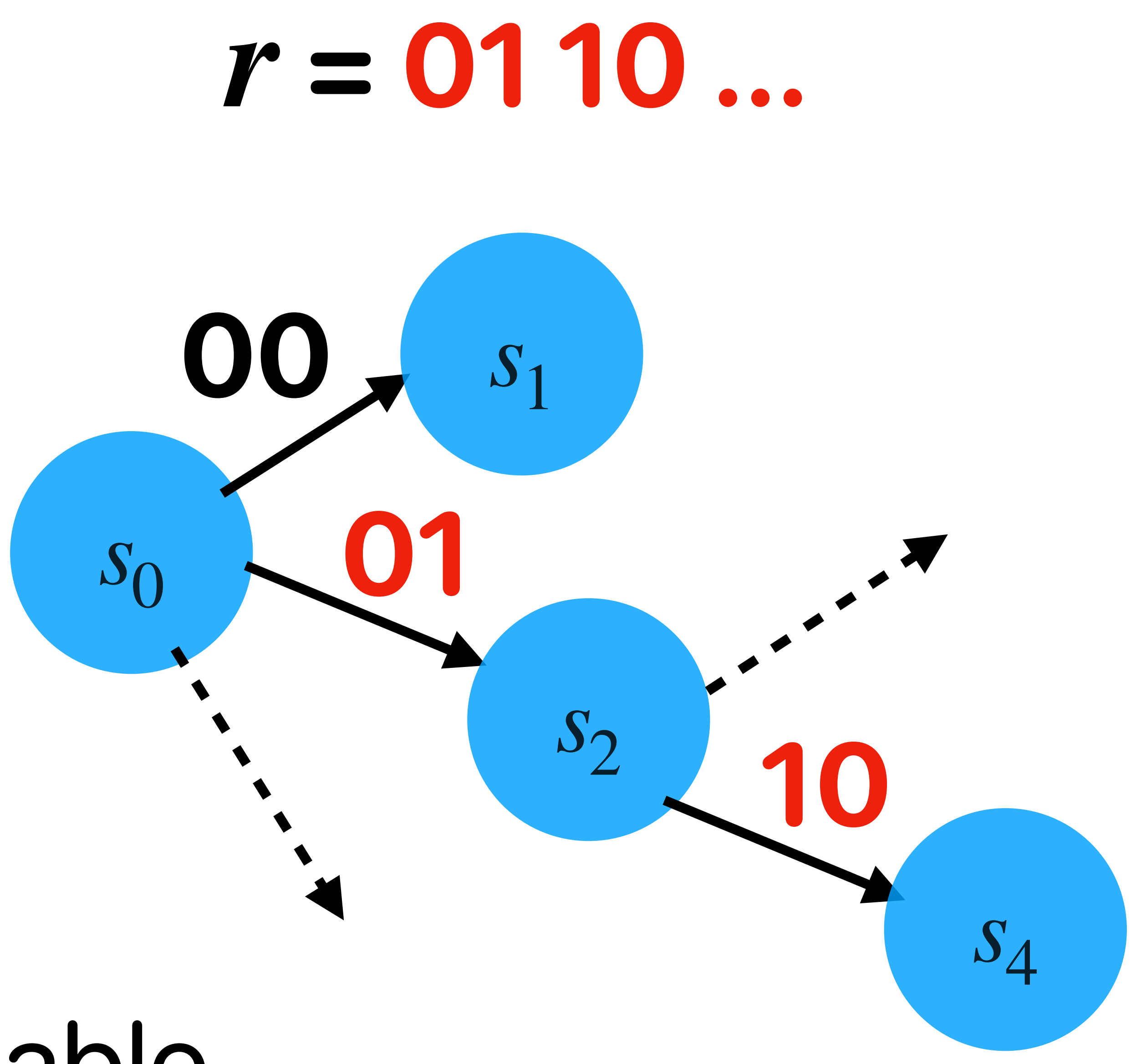
32

$r = 01\,10\ \ldots$



Preserving Properties via Nisan's PRG

- Consider the following algorithm:
  - Uses w bits of space to store its state ($\leq 2^w$ states)
  - Makes a single-pass on the uniform random string
  - Updates its state according to an arbitrary state transition table



- Nisan gave a PRG to "fool" such algorithms by replacing a fully random string with pseudorandom string generated using a **short uniform random seed**
- Indyk gave a recipe to fool turnstile streaming algorithms using such PRGs

Nisan's PRG