Streaming Algorithms

- Streaming Model of Computation
- Streaming Algorithms and DFA
- Stream: Motivation and Applications
- Synopsis: Sliding Window, Histogram, Wavelets
- Sampling from Stream: Reservoir Sampling
- Linear Sketch
- Count-Min Sketch
- AMS Sketch

Imdad ullah Khan

DFA and Streaming Algorithms

Streaming Algorithms can simulate any DFA using logarithmic bits

If L is a language recognized by a DFA D on at most m states, then L is recognized by a streaming algorithm using at most $\log m$ bits of space

We give a streaming algorithm that simulates $D = (Q, \Sigma, q_0, \delta, F)$

Algorithm Streaming Algorithm to simulate $D =$ on string $w = w_1 w_2 \dots w_n$

```
q ← q_0                          ▷ |Q| = m ⟹ q is a log m-bit integer
for i = 1 → n do
    q ← δ(q, w_i)                ▷ δ function can be provided as a lookup table
if q ∈ F then                    ▷ Search for q in F
    Accept
else
    Reject
```

Runtime of the algorithm depends on the data structure of δ and F but space consumed is one integer ($\log m$ bits)

Can DFA simulate Streaming Algorithms?

Clearly, No! $\exists$ non-regular languages recognizable by streaming algorithms

However, DFA's are not totally powerless!

For a language $L \subseteq \Sigma^*$ and an integer $n \geq 0$, let

- $L_n \ = \ \{w \in L : |w| = n\}$
- $L_{\leq n} \ = \ \{w \in L : |w| \leq n\}$

$$L = \{0, 10, 100, 110, 0110, 0100, 1010, 1000, 1110\}$$

- $L_1 = \{0\}$
- $L_{\leq 1} = \{0\}$
- $L_2 = \{10\}$
- $L_{\leq 2} = \{0, 10\}$
- $L_3 = \{100, 110\}$
- $L_{\leq 3} = \{0, 10, 100, 110\}$

Can DFA simulate Streaming Algorithms?

Clearly, No! $\exists$ non-regular languages recognizable by streaming algorithms

However, DFA's are not totally powerless!

Theorem: If $L \subseteq \Sigma^*$ is recognized by a streaming algorithm A using $f(n)$ bits of space on any string of length at most n. Then for every n there is a DFA D with at most $2^{f(n)}$ states, with $L(D)_{\leq n} = L_{\leq n}$

We construct a DFA $D = (Q, \Sigma, q_0, \delta, F)$ from A

- Q : all possible $2^{f(n)}$ values of A's memory
- Σ : alphabet of A or L
- q_0 : initial value of A's memory
- δ : mimics how A's memory changes on σ
- F : states corresponding to A's memory values where A accepts at the end of string

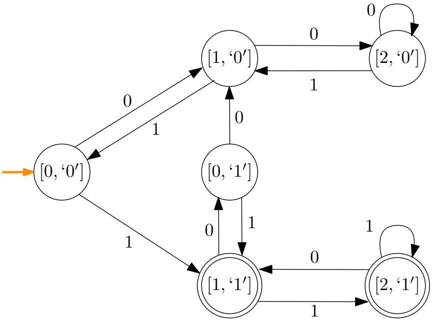
DFA simulating Streaming algorithm to recognize 1-DOMINANT$_{\leq 2}$

Algorithm Streaming Algorithm A for 1-DOMINANT($w = w_1 w_2 \dots w_n$)

$C \leftarrow 0$
$x \leftarrow 0$ ▷ C is int, x is a bit
for $i = 1 \rightarrow n$ **do**
  if $C = 0$ **then**
    $C \leftarrow 1$
    $x \leftarrow w_i$
  else if $C \neq 0$ AND $x = w_i$ **then**
    $C \leftarrow C + 1$
  else
    $C \leftarrow C - 1$
if $C > 0$ AND $x = 1$ **then**
  Accept
else
  Reject

Memory values of A
States of DFA

[C, 'x']



This DFA agrees with A on strings in L of length ≤ 2