\(\Delta\)-coloring in the graph streaming model

Pankaj Kumar  Parth Mittal
Mentor: Dr Sepehr Assadi

REU 2020 \(^1\), Rutgers University

\(^1\)Supported by CoSP, a project funded by European Union’s Horizon 2020 research and innovation programme, grant agreement No. 823748.
The streaming model

- Motivation: Many problems have interesting instances where conventional algorithms use too much space.
The streaming model

- Motivation: Many problems have interesting instances where conventional algorithms use too much space.
- The setting:
  - We have a small number of passes (typically 1 or $O(1)$) over the input.

Examples:

- Median-finding ($O(p)$ pass $\tilde{O}(n^{1/p})$ space algorithms are known).
- Counting distinct elements ($1 + \epsilon$)-approx in $O(1/\epsilon^2 \cdot \log n)$ space with constant probability of success known.)
The streaming model

- Motivation: Many problems have interesting instances where conventional algorithms use too much space.
- The setting:
  - We have a small number of passes (typically 1 or $O(1)$) over the input.
  - Memory is limited to some inconvenient amount
The streaming model

- Motivation: Many problems have interesting instances where conventional algorithms use too much space.
- The setting:
  - We have a small number of passes (typically 1 or $O(1)$) over the input.
  - Memory is limited to some inconvenient amount.
- Examples:
  - Median-finding ($O(p)$ pass $\tilde{O}(n^{1/p})$ space algorithms are known).
The streaming model

- Motivation: Many problems have interesting instances where conventional algorithms use too much space.
- The setting:
  - We have a small number of passes (typically 1 or $O(1)$) over the input.
  - Memory is limited to some inconvenient amount
- Examples:
  - Median-finding ($O(p)$ pass $\tilde{O}(n^{1/p})$ space algorithms are known).
  - Counting distinct elements ($(1 + \epsilon)$-approx in $O(1/\epsilon^2 \cdot \log n)$ space with constant probability of success known).
Graph problems in the streaming model

- Many graphs are too large to store in memory (e.g., the internet)
Graph problems in the streaming model

- Many graphs are too large to store in memory (e.g., the internet)
- Can we design algorithms that work with sub-linear (in the number of vertices) space?

No, even connectivity requires $\Omega(n)$ space.

Semi-streaming model: Allow $O(n \cdot \text{polylog}(n))$ space.
Graph problems in the streaming model

- Many graphs are too large to store in memory (e.g., the internet)
- Can we design algorithms that work with sub-linear (in the number of vertices) space?
- No, even connectivity requires $\Omega(n)$ space.
Graph problems in the streaming model

- Many graphs are too large to store in memory (e.g., the internet)
- Can we design algorithms that work with sub-linear (in the number of vertices) space?
- No, even connectivity requires $\Omega(n)$ space.
- Semi-streaming model: Allow $O(n \cdot \text{polylog}(n))$ space.
Graph problems in the streaming model

- Many graphs are too large to store in memory (e.g., the internet)
- Can we design algorithms that work with sub-linear (in the number of vertices) space?
- No, even connectivity requires $\Omega(n)$ space.
- Semi-streaming model: Allow $O(n \cdot \text{polylog}(n))$ space.
- A good survey is [2]
Let $G = (V, E)$ be a graph, then $c : V \rightarrow C$ is a proper coloring if for any $\{u, v\} \in E$, $c(u) \neq c(v)$.
Let $G = (V, E)$ be a graph, then $c : V \rightarrow C$ is a proper coloring if for any $\{u, v\} \in E$, $c(u) \neq c(v)$.

For $G$ with maximum degree $\Delta$, $\Delta + 1$ coloring is easy (assign arb. color to some vertex, then color rest “greedily”).

If $G$ is not a clique or an odd-cycle, then it can be $\Delta$-colored (Brook's Theorem).
Δ-coloring

Let $G = (V, E)$ be a graph, then $c : V \rightarrow C$ is a proper coloring if for any $\{u, v\} \in E$, $c(u) \neq c(v)$.

For $G$ with maximum degree $\Delta$, $\Delta + 1$ coloring is easy (assign arb. color to some vertex, then color rest “greedily”).

If $G$ is not a clique or an odd-cycle, then it can be $\Delta$-colored (Brook’s Theorem).
Our project

- It is known that randomized $(\Delta + 1)$-coloring is possible in $\tilde{O}(n)$ space [1]
Our project

- It is known that randomized \((\Delta + 1)\)-coloring is possible in \(\tilde{O}(n)\) space \([1]\).
- With randomization, can we \(\Delta\)-color graphs in \(\tilde{O}(n)\) space in \(O(1)\) passes?
References

Sublinear algorithms for $(\Delta + 1)$ vertex coloring.

A. McGregor.
Graph stream algorithms: a survey.