CSE 473 Algorithms: Lecture 12 (2022-03-29)

Logistics:
- Problem set due Fri
- Exam 3 next Fri

Today: Randomized alg.

Q: When does randomness help algorithm design?
A: Almost always. If Pr ≤ 1

- Breaking symmetry
- Divide and conquer
- Dynamic programming
- Randomized divide
- Randomized greedy
- Randomized search
- Randomized rounding

Q: What is a min cut problem?
A: The capacity of a min cut problem is to compute \( \min \{ w(C) \} \)

Q: What is a min cut in a directed undirected graph?
A: The min cut of a graph is \( (S, T) \) where

\[ C = \sum_{(u, v) \in E} w(u, v) \]

Then to my cut \( C \) with \( V = A \cup B \), \( x \in A \), \( y \in B \), \( |C| = |C| \)

Directed min cut in deterministic \( O(n^2m) \) time

For \( t = v, y \), arbitrary choice of \( w(\cdot, \cdot) \) in \( G' \)

Return min of \( O(n^2m) \)
Theorem: Consider a graph $G = (V, E)$ with $\delta(G)$.

1. **Random Correlation** ($H$)
   - $H$ is a multigraph on $V$.
   - $V$ is a set of vertices $V = V_1, \ldots, V_k$.
   - $V_i$ is a set of vertices $V_i$.
   - $e = (u, v)$ if $u \in V_i$ and $v \in V_j$ with $i \neq j$.
   - $e = (u, u)$ if $u \in V$.
   - $e = (u, v)$ if $u, v \in V$.

2. **Minimum Cut** $\delta(G)$
   - $\delta(G)$ is the minimum cut set.
   - $\delta(G)$ is a set of edges that separate the graph into two parts.
   - $\delta(G)$ is a set of vertices that are incident to the cut edges.
   - $\delta(G)$ is a set of edges that have one endpoint in each part of the cut.

Proof:

- Consider a graph $G = (V, E)$ with $\delta(G)$.
- The minimum cut set $\delta(G)$ is a set of edges that separate the graph into two parts.
- $\delta(G)$ is a set of vertices that are incident to the cut edges.
- $\delta(G)$ is a set of edges that have one endpoint in each part of the cut.

**Optimal Solution**

- The optimal solution is a set of edges that minimize the cut.
- The optimal solution can be found using the Ford-Fulkerson algorithm.

**Random Correlation Algorithm**

- The random correlation algorithm is simple.
- The algorithm yields a better solution than the optimal solution.
- The random correlation algorithm is a way to approximate the optimal solution.

**Diagram**

- The diagram shows a random correlation $G = (V, E)$.
- The random correlation algorithm is applied to find the minimum cut set.
- The minimum cut set is found by finding the minimum number of edges that separate the graph into two parts.
- The minimum cut set is a set of edges that minimize the cut.

**Formulas**

- $\delta(G)$ is the minimum cut set.
- $\delta(G)$ is a set of edges that separate the graph into two parts.
- $\delta(G)$ is a set of vertices that are incident to the cut edges.
- $\delta(G)$ is a set of edges that have one endpoint in each part of the cut.

**Notations**

- $G = (V, E)$ is a graph.
- $\delta(G)$ is the minimum cut set.
- $V$ is a set of vertices.
- $E$ is a set of edges.
- $e = (u, v)$ is an edge between vertices $u$ and $v$.
\[ E : \text{no edge of } C \text{ is connected in } T \]\\
\[ L : T \text{ is a } S \text{-tree, where } S \text{ is any edge of } C. \]

If the edge is not in \( C \), then \( A \cap B = \emptyset \) and \( A \cup B = V \setminus \{v, w\} \).

If the edge is in \( C \), then \( A \cap B = \{v, w\} \) and \( A \cup B = \emptyset \).

Similarly, \( C \) can be connected in \( T \).}

\[ E : \text{success probability.} \]

\[ C_{ij} = \begin{cases} 0 & \text{if } i = j, \\
1 & \text{if } i \neq j \text{ and } i, j \in V. \end{cases} \]

For an \( n \times n \) upper triangular matrix, it is possible to select \( n-1 \) rows and columns to form a non-zero submatrix.

\[ \text{Select a non-zero submatrix} \]
Lemma I: \[ E_1 = m \]

Proof: \[ E_1 = \min \{ \text{no edge of } C \text{ connected in } \chi \} \]

- If \( |C| = 1 \), then \( \chi \) is a tree, and it is connected in \( \chi \).

- If \( |C| > 1 \), then \( \chi \) is a forest, and it is connected in \( \chi \).

Thus, \( E_1 \) is the minimum number of edges necessary for \( \chi \) to be connected.

\[ E_1 = 1 - \frac{1}{|C|} \]

\[ E_1 = \frac{n^n - 1}{n - 1} \]

\[ E_1 = \frac{n^n - 1}{n - 1} \]

- For \( n \geq 2 \), \( E_1 \) is the minimum number of edges necessary for \( \chi \) to be connected.

\[ E_1 \geq 1 - \frac{2}{(n - 1)} \]

- For \( n \geq 2 \), \( E_1 \) is the minimum number of edges necessary for \( \chi \) to be connected.

\[ E_1 \geq 1 - \frac{2}{(n - 1)} \]

\[ E_1 \geq 1 - \frac{2}{(n - 1)} \]

- For \( n \geq 2 \), \( E_1 \) is the minimum number of edges necessary for \( \chi \) to be connected.

\[ E_1 \geq 1 - \frac{2}{(n - 1)} \]

- For \( n \geq 2 \), \( E_1 \) is the minimum number of edges necessary for \( \chi \) to be connected.

\[ E_1 \geq 1 - \frac{2}{(n - 1)} \]

\[ E_1 \geq 1 - \frac{2}{(n - 1)} \]

\[ E_1 \geq 1 - \frac{2}{(n - 1)} \]

Thus, \( E_1 \) is the minimum number of edges necessary for \( \chi \) to be connected.