CS289a: Great Theory Hits of 21st Century

Tingfeng Xia

Winter 2023

Notes reorganized from `https://hackmd.io/@raghum/greathits`.

Contents

Chapter 1

Undirected s-t Connectedness

1.1 Computing Resources

Four main computing resources that we consider as limited (and measure the performance of our algorithms against)

- Time
- Memory
- Randomness
- Communication

1.2 Problem Statement

- **Input**: Graph $G = (V, E)$; with source and target marked as s, t
- **Output**: YES iff s and t are connected, NO otw.

Above is the "traditional" definition of $s-t$ connectivity which we can solve with a vanilla BFS or DFS. This will take $\mathcal{O}(|V| + |E|)$ and $\mathcal{O}(|V|)$ extra bits of space / memory. The question is then, can we solve the same problem with sub-linear extra memory usage.

Proposition 1.1 *There is a randomized algorithm with* $5 \log |V|$ *bits of additional memory (directed and undirected graphs).*

Proposition 1.2 (Omer Reingold, 2005) *There is a log space ($\mathcal{O}(\log |V|)$) algorithm* ***(deterministic)*** *for undirected graphs.* [1.2.1]

[1.2.1] first great hit ...

It is yet unknown if we can achieve log space for directed graphs (with deterministic algorithm). The best known algorithms runs with $O(\log |V|)^{3/2}$ bits of memory. Why is this so challenging?

Proposition 1.3 *If divided $s - t$ connectivity can be solved with $O(\log |V|)$ extra bits of memory (without randomness), then any randomized algorithm can be made deterministic at the expenses of a constant factor increase in memory.*

1.3 Randomized Algorithm for Connectivity

Algorithm 1.1 (Random Walk Algorithm for Connectivity) *Here is the algorithm*

- *steps* $\leftarrow 0$
- *current* $\leftarrow$ *s; target* $\leftarrow$ *t*
- *while steps* $< T$
  - *current* $\leftarrow$ *random neighbor of current*
  - *if current* $==$ *target return YES*
- *return NO*

The total memory for this algorithm is

$$2 \log N + \log T \leq 5 \log N \tag{1.3.1}$$

extra bits, assuming we can get random neighbor.

Proposition 1.4 (Alenilaus, 80s) *If $T = 100N^3$ steps, then Pr[Algorithm wrong] $< \frac{1}{3}$*

which can improved to arbitrary accuracy by repeating the algorithm. Algorithms of this nature can perform bad on graphs known as "Lollipop Graphs" and even worse a "Dumbell Graph"

1.4 Log Space USTCON

Here we highlight the progression in space complexity in various papers

- **Nisan, 92**: Space $O(\log^2 N)$, time $N^{O(1)}$ algorithm... improved to $O(\log^{4/3} N)$ in space.
- **Reingold, 05**: Space $O(\log N)$, time $N^{O(1)}$ algorithm.
- **Trifornov, 05**: Space $O((\log N)(\log \log N))$ algorithm.

1.5 Spectral Graph Theory

Consider an undirected graph $G = (V, E)$,

Definition 1.1 (Degree) *Degree of a vertex v is the number f edges v is connected to.*

Definition 1.2 (Regular) *Graphs is "regular" if all vertices have same degree.*

Definition 1.3 (Adjacency Matrix) *$A(G)$ is a symmetric matrix where $A(G)_{ij} = 1$ if $\{i, j\}$ is an edge, 0 otw.*

Definition 1.4 (Normalized Adj Matrix) *If G is regular and has degree D, then the normalized adjacency matrix is defined as*

$$M(G) \equiv \frac{A(G)}{D} \tag{1.5.1}$$

Lemma 1.1 *If G is regular, then 1 is an eigenvalue of $M(G)$. And $\mathbf{v}_1 = \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}^\top$ is an eigenvector with eigenvalue 1.*

Proposition 1.5 (Eigenvalues of Regular Graphs) *If G is regular, then all eigenvalues of $M(G)$ have magnitude ≤ 1.*

Proof: WLOG assume x_3 is the largest entry in the vector $\mathbf{x}$, then

$$\begin{aligned}
\lambda|x_3| &= |M_{31}x_1 + M_{32}x_2 + \dots + M_{3N}x_N| && (1.5.2)\\
&\leq M_{31}|x_3| + M_{32}|x_3| + \dots + M_{3N}|x_3| && (1.5.3)\\
&= (M_{31} + \dots M_{3N})|x_3| && (1.5.4)\\
&= 1|x_3| && (1.5.5)
\end{aligned}$$

Thus, $\lambda \leq 1$. ■

Proposition 1.6 (Connectedness and Matrices) *Regular $G = (V, E)$ is connected if and only if the only eigenvector with eigenvalue 1 for $M(G)$ is the all 1 vector.*

Might appear on exam 1

Proposition 1.7 (Eigenvalues of a Regular Graph) *If G is regular, then the eigenvalues of $M(G)$ are*

Might appear on exam 1

$$1 = \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N \tag{1.5.6}$$

Proposition 1.8 *G is connected and regular if and only if*

Might appear on exam 1

$$\max(|\lambda_2|, |\lambda_3|, \dots, |\lambda_N|) \leq 1 \tag{1.5.7}$$

Proposition 1.9 (Eigenvalues of D-Regular Graphs) *If G is a D-regular graph, then*

- *1 is an eigenvalue of $M(G)$, and*
- *all eigenvalues of $M(G)$ are at most 1 in absolute value*

Definition 1.5 (Self Loops) *We add connections from each node in the graph to themselves. In the matrix representation, we set $G_{ii} = 1, \forall i$.*

Definition 1.6 (Second Largest Eigenvalue) *... denoted as $\lambda(G)$ or $\lambda_2(G)$.*

Lemma 1.2 *If G is D-regular and **has self loops**, then G is connected if and only if $\lambda(G) < 1$.*

d later

Proof: We first show G is disconnected implies $\lambda(G) = 1$ (via contrapositive).

Definition 1.7 (Spectral Gap) *Spectral Gap of a D-regular graph G is defined as*

$$\textit{Spectral Graph} \equiv 1 - \lambda(G) \tag{1.5.8}$$

Lemma 1.3 *If G is a D-regular connected graph with self-loops, then*

$$\lambda(G) \leq 1 - \frac{1}{2D^2 \cdot N^2} \tag{1.5.9}$$

Definition 1.8 *We say a graph G is (N, D, λ) if it has N vertices, D regular and $\lambda(G) \leq \lambda$.*

1.6 Easy Cases