Plan for this lecture is to start on random walks. The problem that we will use to introduce this concept is computing a maximum matching in \(d\)-regular bipartite graphs.

This is more of a ‘rounding’ problem: a fractional solution is easy: just put \(1/d\) on each edge. There is an \(O(m)\) algorithm for this by Cole, Ostin, and Schirra \cite{COS01}. Their result is based on Eulerian tours, and finding one matching in an odd-degree graph.

There is also a naive algorithm. An augmenting path is a path starting and ending at unmatched vertices and alternating between matched and unmatched edges. We can show that a matching is not maximum if and only if there is an augmenting path.

By Hall’s matching theorem, every \(d\)-regular bipartite graph has a perfect matching. The main result that we will show, due to Goel, Kapralov, and Khanna \cite{GKK13}, is that on \(d\)-regular bipartite graphs, random walk based augmenting paths is actually really fast. Specifically we will show that when \(k\) edges are matched, the expected number of back edges on a random walk from \(s\) to \(t\) is at most

\[
\frac{n}{n-k} - 1.
\]

That is, we can find an augmenting path in sublinear time: we don’t even look at the entire graph. The total cost from such a bound is then

\[
\leq \sum_{i=0}^{n-1} \frac{n}{n-i} = n \sum_{i=1}^{n} \frac{1}{i} = O(n \log n).
\]

1 Analyzing Random Walks

Let \(G = (A, B, E)\) be a \(d\)-regular bipartite graph, and let \(M\) be a partial matching that has \(k\) units matched. Let \(A_M, B_M\) be matched vertices in \(A\) and \(B\), \(A_U\) and \(B_U\) be unmatched vertices in \(A\) and \(B\) respectively. The matching graph corresponding to \(M\) is then defined to be the directed path \(H\) obtained by transforming \(G\) as described below:

1. Orient unmatched edges from \(A\) to \(B\) and matched edges from \(B\) to \(A\).
2. Add a vertex \(s\) connected to each vertex in \(A_U\), directed out of \(s\).
3. Add a vertex \(t\) connected to each vertex in \(B_U\), directed to \(t\).

Any \(s-t\) path in the graph \(H\) defines an augmenting path in \(G\) with respect to the matching \(M\). The core of our algorithm is a random walk on the graph \(H\) starting at the
vertex $s$, in which an outgoing edge is chosen uniformly at random. Once $t$ is reached, the matching $M$ is augmented. We repeat this process until we obtain a perfect matching.

Let $x(u) =$ expected number of back edges starting at $u$ until we get to the sink $t$. Our goal is to bound the expected number of back edges starting at $s$ until we get to $t$

$$\frac{1}{n-k} \sum_{a \in A_U} x(a).$$

which is proportional to the number of steps it takes for a random walk starting at $s$ to reach $t$ in graph $H$. From the construction of $H$, we have

- vertex $b \in B$:  
  - unmatched: $x(b) = 0$.  
  - matched: $x(b) = 1 + x(M(b))$.

- vertex $a \in A$:  
  - unmatched: $x(a) = 1/d \sum_{b \in E} x(b)$.  
  - matched: 
    $$x(a) = \frac{1}{d-1} \left[ \sum_{b \in E \setminus M} x(b) - x(M(a)) \right] = \frac{1}{d-1} \left[ \sum_{b \in E} x(b) - (1 + x(a)) \right].$$

Hence $x(a) = -1/d + 1/d \sum_{b \in E} x(b)$.

- Summing over all $a \in A$:
  $$\sum_a x(a) = -\frac{k}{d} + \sum_b x(b),$$

- Plugging in $\sum_{b \in B_M} x(b) = k + \sum_{b \in B_M} x(M(b))$ gives:
  $$\sum_a x(a) = -\frac{k}{d} + k + \sum_{a \in A_M} x(a),$$

  or $\sum_{a \in A_U} x(a) = -k/d + k$.

- We go to one of the $n-k$ vertices in $A_U$ randomly on the first step. The expected progress is
  $$\frac{1}{n-k} (-k/d + k) \leq \frac{k}{n-k} = \frac{n}{n-k} - 1.$$
2 More Hand-Wavy Analysis

Construct the residue graph corresponding to $M$ as follows:

1. Introduce super source $s$ and super sink $t$;
2. Connect $s$ to each node in $A_U$ by $d$ edges, same for each node in $B_U$ to $t$.
3. Contract each $(u, v) \in M$ into a supernode;
4. Contract $s, t$ into $s^*$.

Consider a random walk in residue graph starting at $s^*$. Note that for any vertex $v$, the in-degree of $v$ is equal to its out-degree, and thus the residue graph is a directed, Eulerian graph. The stationary distribution for the random walk on such a graph is uniform distribution on each edge. Also, the expected return time is equal to the inverse of the stationary measure.

Note that

$$d^{-v} = \begin{cases} 
    d(n-k), & \text{if } v = s^*; \\
    d, & \text{if } v \in A_U \cup B_U; \\
    d - 1, & \text{otherwise}.
\end{cases}$$

The total number of edges in the residue graph is the sum of out-degrees of all vertices

$$d(n-k) + k(d-1) + d \cdot 2(n-k) = k(d-1) + 3d(n-k).$$

Note that a random walk starting at $s^*$ returns to $s^*$ when one of $d(n-k)$ in-edges of $s^*$ is reached. Thus the stationary measure of $s^*$ is

$$\pi_{s^*} = \frac{d(n-k)}{k(d-1) + 3d(n-k)}.$$

Therefore the expected time to return to $s^*$ is

$$\frac{1}{\pi_s} = \frac{k(d-1) + 3d(n-k)}{d(n-k)} \leq 3 + \frac{k}{n-k} = 2 + \frac{n}{n-k}.$$

References
