Perfect matchings in $O(n \log n)$ time in regular bipartite graphs

Research project for computational optimization

Given by: Ashish Goel, Michele Kapralov, and Sanjeev Khanna

Presented by: Qing Li

April 26, 2011
Outline

i. Introduction.

ii. Matching in $d$-regular bipartite graphs.

iii. An $\Omega(nd)$ lower bound for deterministic algorithms.

iv. Conclusion.
What is $d$–regular bipartite graph?

A bipartite graph $G = (P, Q, E)$ with vertex set $P \cup Q$ and edge set $E \subset P \times Q$ is said to be $d$-regular if every vertex has the same degree $d$. We use $m = nd$ to denote the number of edges in $G$ and $n$ to represent the number of vertices in $P$.
Preliminary

- The graph is presented mainly in the adjacency array format, i.e., for each vertex, its $d$ neighbors are stored in an array.
- An augmenting path is a path which starts and ends at an unmatched vertex, and alternately contains edges that are outside and inside the partial matching.
- *Alternating random walk* on $G$ with respect to $M$: starts at a uniformly random unmatched vertex $u \in P$ and randomly returns $u$’s uniformly random unmatched vertex in $Q$. 

Given by: Ashish Goel, Michele Kapralov, and Sanjeev Khanna

Perfect matchings in $O(n \log n)$ time in regular bipartite graphs
An example

1 is matched to 4

Augmenting path: 5 − 1 − 4 − 2

Alternating random walk: 3 − 6

Figure 2: An example
The basic algorithm

**Input:** A $d$-regular bipartite graph $G = (P, Q, E)$ in adjacency array format.

**Output** A perfect matching of $G$.

1. Set $j := 0$, $M_0 := \emptyset$.
2. Run the alternating random walk starting from a random unmatched vertex in $P$ until it hits an unmatched vertex in $Q$.
3. Denote the augmenting path obtained by removing possible loops from the sequence of steps taken by the walk $p$. Set $M_{j+1} := M_j \Delta p$.
4. Set $j := j + 1$ and go to step 2.
Illustration of the basic algorithm

1 is matched to 4

Augmenting path: 5 – 1 – 4 – 2

Alternating random walk: 3 – 6

Figure 2: An example
Expected running time analysis

Lemma

Let $G = (P, Q, E)$ be a $d$-regular bipartite graph and let $M$ be a partial matching that leaves $2k$ vertices unmatched. Then the expected number of steps before the alternating random walk in $G$ reaches an unmatched vertex is at most $1 + n/k$.

Proof.

1. Orient edges of $G$ from $P$ to $Q$;
2. Contact each pair $(u, v) \in M$ into a supernode;
3. Add a vertex $s$ connected by $d$ parallel edges to each unmatched node in $P$, directed out of $s$.
4. Add a vertex $t$ connected by $d$ parallel edges to each unmatched node in $Q$, directed into $t$. 

Given by: Ashish Goel, Micheal Kapralov, and Sanjeev Khanna

Perfect matchings in $O(n \log n)$ time in regular bipartite graphs
An example

Figure 1: original graph and matching graph

- alternating random walks in $G$ with respect to $M$ corresponds to random walks in $H(G, M)$.
the expected number of steps before the random walk reaches \( t \) is at most
\[
\sum_{j \in V(H), j \neq s, t} \frac{\deg(j)}{dk} \leq \frac{(n - k) + 2k}{k} = 1 + \frac{n}{k}.
\]
Thus we can prove the following theorem

**Theorem**

*The basic algorithm finds a matching in \( d \)-regular bipartite graph \( G = (P,Q,E) \) in expected time \( O(n \log n) \).*

**Proof.**

\[
\sum_{j=0}^{n-1} 1 + n/(n - j) = n + nH_n = O(n \log n),
\]
where \( H(n) := 1 + 1/2 + 1/3 + \ldots + 1/n \) is the \( n \)-th Harmonic number.
Truncated random walks

**Input:** A $d$-regular bipartite graph $G = (P, Q, E)$ in adjacency array format.

**Output:** A perfect matching of $G$.

1. Set $j := 0$, $M_0 = \emptyset$.
2. Repeatedly run alternating random walks for $2(1 + \frac{n}{n-j})$ steps until a successful run is obtained.
3. Denote the augmenting path obtained by removing possible loops from the sequence of steps taken by the walk by $p$. Set $M_{j+1} := M_j \Delta p$.
4. Set $j := j + 1$ and go to step 2.
Analysis of the truncated random walks

Theorem

There exists a randomized algorithm for finding a perfect matching in $d$-regular bipartite graph $G = (P, Q, E)$ given in adjacency array representation, and takes $O(n \log n)$ time both in expectation as well as with high probability.
The canonical graph contains the perfecting matchings, thus if every graph formed in the process of probing is subgraph of it, finally we will find the perfecting matching.
Overview

- Initially, algorithm $A$ knows nothing about the graph $G$. $A$ probes and the adversary $\mathcal{A}$, trying to maximize the running time, tells something about the graph $G$. $A$ forms a new subset graph $G'$ of $G$ according to the information.
- If every time $G'$ is the subgraph of the canonical graph, then we can find out the perfecting matching.
- A vertex $u \in P \cup Q$ is free if its degree in $G'$ is strictly smaller than $d$. 

Figure: 6 overview of a deterministic algorithm
Lemma

Let $G_r(P \cup \{t\}, Q \cup \{s\}, E_r)$ be any bipartite graph such that

1. The vertex $s$ is connected to $d$ distinct vertices in $P_1$ and the vertex $t$ is connected to $d$ distinct vertices in $Q_2$.
2. all other edges in $G_r$ connect a vertex $P_i$ to a vertex in $Q_i$ for some $i \in \{1, 2\}$.
3. degree of each vertex in $G_r$ is at most $d$, and
4. there exist at least $(d + 1)$ free vertices each in both $Q_1$ and $P_2$.

Also, let $u, v$ be a pair of vertices such that $u \in P_i$ and $Q_i$ for some $i \in \{1, 2\}$, and $(u, v) \notin E_r$. Then there exists a canonical graph $G(U \cup \{t\}, Q \cup \{s\}, E) \in \mathcal{G}(d)$ such that that $E_r \cup (u, v) \subset E$ iff both $u$ and $v$ have degree strictly less than $d$ in $G_r$. 

Given by: Ashish Goel, Micheal Kapralov, and Sanjeev Khanna

Perfect matchings in $O(n \log n)$ time in regular bipartite graphs
When to stop?

- When we can not form a subgraph of the canonical graph, meaning we know all about the canonical graph and the perfecting matching, we can stop.
- Check the conditions: only (4) is possible to fail.
- Every probe adds one edge and thus one degree. Initially $P$ has $2d$ vertices, and every vertex has at most $d$ degrees, thus at least $d^2$ probes can violate (4).
There exists an $O(n \log n)$ expected time algorithm for finding a perfect matching in a $d$-regular bipartite graph $G=(P,Q,E)$ given in adjacency array representation.

For any $1 \leq d \leq n/8$, there exists a family of $d$-degree graphs on which any deterministic algorithm for finding a perfect matching requires $\Omega(nd)$ time.