Notes on Simple Analysis for Parallel Maximal Independent Set and Maximal Matching Algorithms

Julian Shun

1 Maximal Independent Set (MIS)

For a graph $G = (V, E)$ we use $N(V)$ to denote the set of all neighbors of vertices in V. A maximal independent set $U \subset V$ is thus one that satisfies $N(U) \cap U = \emptyset$ and $N(U) \cup U = V$. We use $N(v)$ as a shorthand for $N(\{v\})$ when v is a single vertex. We use $G[U]$ to denote the **vertex-induced subgraph** of G by vertex set U, i.e., $G[U]$ contains all vertices in U along with edges of G with both endpoints in U. A simple parallel algorithm due to Luby [4] for computing the maximal independent set is as follows:

Algorithm 1 Luby's algorithm for MIS

1: **procedure** $\text{MIS}(G = (V, E))$
2:     **if** $|V| = 0$ **then return** $\emptyset$
3:     **else**
4:         randomly assign unique priorities π_v to each $v \in V$
5:         let W be the set of vertices in V with higher priority than all
6:             of its neighbors (i.e. vertices v such that $\forall u \in N(v), \pi_v > \pi_u$)
7:         $V' = V \setminus (W \cup N(W))$
8:         **return** $W \cup \text{MIS}(G[V'])$

Theorem 1.1. *Each recursive call to* MIS *removes half of the number of edges in G in expectation.*

Proof. Here is a simple analysis due to Métivier et. al. [5]:

Consider an edge (u, v) in G. Define the indicator variable X_u to be the event that u's priority is greater than that of all of its neighbors and all of v's neighbors (excluding u itself). Note that if this event happens, u will be included in the MIS, and all edges incident to u and v will be removed. For the purpose of this analysis we say that if X_u happens, then u *preemptively removes* v and the edges incident to v. Note that any vertex can be preemptively removed at most once, and any edge (u, w) can only be preemptively removed twice, once when u is preemptively removed and once when w is preemptively removed. Note that $Pr(X_u = 1) \geq 1/(d(u) + d(v))$ (it is an inequality becasue u and v may share some neighbors).

Since we double counted the edge removals, the average number of edges removed is

$$\frac{1}{2} \sum_{(u,v) \in E} (d(v)Pr(X_u = 1) + d(u)Pr(X_v = 1))$$

as $d(v)$ edges are removed if u preemptively removes v and $d(u)$ edges are removed if v preemptively removes u. This quantity is at least

$$\frac{1}{2} \sum_{(u,v)\in E} \left(\frac{d(v)}{d(u)+d(v)} + \frac{d(u)}{d(u)+d(v)} \right) = \frac{1}{2} \sum_{(u,v)\in E} 1 = \frac{|E|}{2}$$

$\square$

Theorem 1.2. *The total expected work of* MIS *is $O(|E|)$ and total expected span is $O(\log^2 |E|)$.*

Proof. Each recursive call does $O(|E|)$ work and decreases the number of edges by at least half, so the total work $O(|E|)$. Since the number of edges decreases by a constant fraction in each round, we have $O(\log |E|)$ rounds. Each round requires $O(\log |E|)$ span for checking the priorities and producing the subgraph. Thus the total span is $O(\log^2 |E|)$. $\square$

2 Maximal Matching

For a graph $G = (V, E)$ we use $N(E)$ to denote the neighboring edges of E (ones that share a vertex). A maximal matching E' is one that satisfies $N(E') \cap E' = \emptyset$ and $N(E') \cup E' = E$. We use $N(e)$ as a shorthand for $N(\{e\})$ when e is a single edge. We use $G[E'']$ to denote the **edge-induced subgraph** of G by edge set E''.

Algorithm 2 Luby's algorithm for MIS

1: **procedure** MATCHING($G = (V, E)$)
2:     **if** $|E| = 0$ **then return** $\emptyset$
3:     **else**
4:         randomly assign unique priorities π_e to each $e \in E$
5:         let W be the set of edges in E with higher priority than all
6:             of its neighboring edges (i.e. edges e such that $\forall e' \in N(e), \pi_e > \pi_{e'}$)
7:         $E' = E \setminus (W \cup N(W))$
8:         **return** $W \cup$ MATCHING($G[E']$)

Theorem 2.1. *Each recursive call to* MATCHING *removes half of the number of edges in G in expectation.*

Proof. The proof of Métivier et. al. [5] can be adapted to the case of MATCHING (see [1]). We say that an edge **wins** if it has higher priority than all of its neighboring edges. If an edge wins it preemptively deletes itself and all of its neighboring edges from the graph. An edge can only be preemptively deleted at most twice, once by a winning edge from each of its endpoints. An edge (u, v) wins with probability $1/(d(u) + d(v) - 1)$, and if it wins it preemptively deletes $d(u) + d(v) - 1$ edges. Therefore, the expected number of edges deleted is at least

$$\frac{1}{2} \sum_{(u,v)\in E} (d(u) + d(v) - 1) \frac{1}{d(u) + d(v) - 1} = \frac{|E|}{2}$$

$\square$

Theorem 2.2. *The total expected work of* MATCHING *is $O(|E|)$ and total expected span is $O(\log^2 |E|)$.*

Proof. The proof is similar to that of MIS. □

Note that this maximal matching algorithm and analysis is much simpler than previous parallel algorithms for maximal matching [2, 3].

References

[1] M. Birn, V. Osipov, P. Sanders, C. Schulz, and N. Sitchinava. Efficient parallel and external matching. In *Euro-Par 2013 Parallel Processing*, pages 659–670. 2013.

[2] A. Israeli and A. Itai. A fast and simple randomized parallel algorithm for maximal matching. *Inf. Process. Lett.*, 22:77–80, February 1986.

[3] A. Israeli and Y. Shiloach. An improved parallel algorithm for maximal matching. *Inf. Process. Lett.*, 22:57–60, February 1986.

[4] M. Luby. A simple parallel algorithm for the maximal independent set problem. *SIAM J. Comput.*, 1986.

[5] Y. Métivier, J. M. Robson, N. Saheb-Djahromi, and A. Zemmari. An optimal bit complexity randomized distributed mis algorithm (extended abstract). In *SIROCCO*, pages 323–337, 2009.